

Industrial communications network with mastership determined by need

Patent Number: ☐ US4319338
Publication date: 1982-03-09
Inventor(s): GRUDOWSKI RAYMOND A; ENGDAHL JONATHAN R
Applicant(s):: ALLEN BRADLEY CO
Requested Patent: ☐ DE3043894
Application: US19790102970 19791212
Priority Number(s): US19790102970 19791212
IPC Classification: G06F3/00
EC Classification: G05B19/05M, G06F13/22B, G06F13/368, G06F13/372,
Equivalents: CA1143028, ☐ GB2064920

Abstract

An industrial communications network includes microprocessor-based interface circuits which each connect a controller such as a programmable controller to a high speed serial data link. Each interface circuit connects to the data link and its associated controller, and each is operable to receive messages on the data link directed to its associated controller. In addition, each interface circuit can assume mastership of the communications network when the existing master generates a poll command indicating it is ready to relinquish mastership. As a result, the communications network will continue to function even though one or more controller or their associated interface circuits become inoperable.

Data supplied from the esp@cenet database - I2

①9 BUNDESREPUBLIK
DEUTSCHLAND



DEUTSCHES
PATENTAMT

⑫ Pat ntschrift
⑪ DE 3043894 C2

②1 Aktenzeichen: P 30 43 894.9-31
②2 Anmeldetag: 21. 11. 80
④3 Offenlegungstag: 19. 6. 81
④5 Veröffentlichungstag
der Patenterteilung: 28. 12. 89

⑤1 Int. Cl. 4:
H 04 L 11/00
H 04 L 25/00
G 06 F 15/46
G 05 B 15/02
G 05 B 19/02

DE 3043894 C2

Innerhalb von 3 Monaten nach Veröffentlichung der Erteilung kann Einspruch erhoben werden

③0 Unionspriorität: ③2 ③3 ③1
12.12.79 US 102970

⑦3 Patentinhaber:
Allen-Bradley Co., Milwaukee, Wis., US

⑦4 Vertreter:
Reichel, W., Dipl.-Ing.; Lippert, H., Dipl.-Ing.,
Pat.-Anwälte, 6000 Frankfurt

⑦2 Erfinder:
Grudowski, Raymond A., South Euclid, Ohio, US;
Engdahl, Jonathan R., Maple Heights, Ohio, US

⑤6 Für die Beurteilung der Patentfähigkeit
in Betracht gezogene Druckschriften:

DE-AS 26 31 052
GB 11 68 476

⑤4 Verfahren zur Steuerung der Nachrichtenübertragung in einem Kommunikationsnetzwerk und
Schnittstellen-Modul für ein derart gesteuertes Kommunikationsnetzwerk

DE 3043894 C2

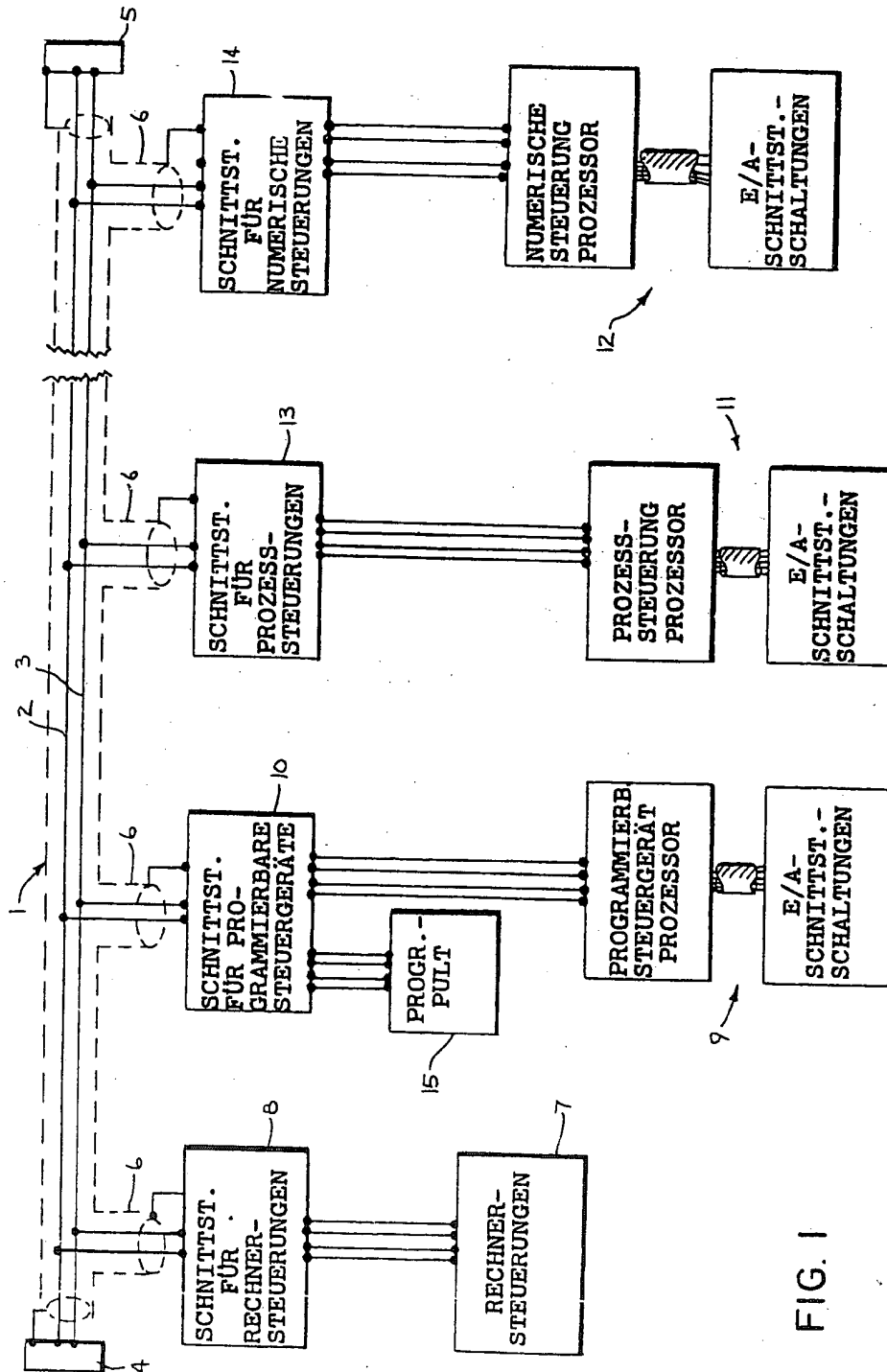


FIG. 1

Beschreibung

Die Erfindung bezieht sich auf ein Verfahren zur Steuerung der Nachrichtenübertragung in einem Kommunikationsnetzwerk mit einer Vielzahl an ein Kabel angeschlossener Schnittstellen-Module gemäß dem Oberbegriff des Patentanspruchs 1 und betrifft ferner einen Schnittstellen-Modul für ein derart gesteuertes Kommunikationsnetzwerk. Ein Verfahren sowie ein Schnittstellen-Modul der oben angegebenen Art sind aus der DE-AS 26 31 052 bekannt. Bevor dieser Stand der Technik näher erläutert wird, sei zum besseren Verständnis der Erfindung folgendes vorausgeschickt.

Für industrielle Steuerungen und Steuergeräte eingesetzte Kommunikations- oder Informationsaustauschnetzwerke haben sich im Laufe der Zeit so entwickelt, daß sie als Stern-, Mehrfachabzweig- oder Ringanordnung ausgebildet sind. Bei der Sternanordnung bildet ein Steuergerät den Mittelpunkt und arbeitet bezüglich des Netzwerks als Hauptsteuergerät. Von diesem Hauptsteuergerät erstrecken sich getrennte Leitungen zu allen anderen Steuergeräten, die Nebensteuergeräte genannt werden. Die Vielfach- oder Mehrfachabzweiganordnung enthält gemäß der US-PS 41 49 144 ebenfalls nur ein einziges Hauptsteuergerät, das mit einem einzigen Datenweg verbunden ist. Die Nebensteuergeräte sind über Leitungen an diesen Datenweg angeschlossen. Diese Leitungen werden von dem Datenweg abgezweigt, und der Verkehr zwischen dem Hauptsteuergerät und einem ausgewählten Nebensteuergerät geschieht über einen "Aufruf"-Prozeß. Bei einem derartigen Mehrfachabzweignetzwerk ist ein Informationsaustausch oder Datenverkehr zwischen Nebensteuergeräten nur über das Hauptsteuergerät möglich. Bei der Ringanordnung ist jedes Steuergerät mit zwei anderen Steuergeräten verbunden. Es liegt somit eine schleifenförmige Anordnung vor. Die Nachrichten werden von einem Steuergerät zum nächsten Steuergerät weitergeleitet. Ein Netzwerkhauptgerät unterbindet die Weiterleitung von Nachrichten, die bereits den vollen Kreis durchlaufen haben und von keinem der Steuergeräte im Ring bestätigt worden sind.

All diese bekannten Kommunikationsnetzwerke haben die Unzulänglichkeit, daß das gesamte System ausfällt, wenn das Hauptsteuergerät gestört ist. Bei der Sternanordnung werden die Nebensteuergeräte voneinander getrennt, wenn das Hauptsteuergerät nicht mehr arbeitet. Bei der Mehrfachabzweiganordnung wird beim Ausfall des Hauptsteuergerätes der gesamte Informations- und Nachrichtenaustausch unterbrochen, da der Aufrufprozeß oder Aufrufvorgang nicht mehr ausgeführt werden kann. Bei der Ringanordnung führt die Störung irgendeines der Steuergeräte zur Unterbrechung der Verbindung. Um diese Unzulänglichkeit zu überwinden, hat man in der Praxis durch Anwendung redundanter Maßnahmen versucht, den Ausfall des gesamten Netzwerks zu verhindern. Diese Lösungsmöglichkeit ist jedoch äußerst kostspielig.

Eine andere Lösungsmöglichkeit besteht in der Anwendung eines Wettbewerbschemas beim Mehrfachabzweignetzwerk. Bei einem solchen System entfällt das Hauptsteuergerät, und statt dessen streiten sich die Steuergeräte um den Zugriff zur Datenverbindung. Wenn ein Steuergerät den Zugriff erlangt hat, kann man zu jedem anderen Steuergerät im Netzwerk eine Nachricht senden. Es sind zahlreiche derartige Wettbewerbsverfahren bekannt geworden. Obgleich sie die Abhängigkeit von einem Hauptsteuergerät zur Aufrechterhaltung der Kommunikation beseitigen, ist kein einziges der Wettbewerbsverfahren universell auf alle Datenverkehrsmuster anwendbar. Darüber hinaus muß man spezielle Hardware hinzufügen, um die Kollision von Nachrichten auf dem Daten Verbindungsweg zu vermeiden. Solche Maßnahmen erhöhen die Kosten und führen zu komplexen Netzwerken.

Aus der bereits zitierten DE-AS 26 31 052 ist ein Datenübertragungssystem mit mehreren Datenverarbeitungseinheiten bekannt, die über je eine Übertragungssteuerung parallel an eine Sammelleitung angeschlossen sind. Über die Sammelleitung werden Daten und/oder Steuersignale übertragen. Dieses bekannte Datenübertragungssystem ist derart ausgebildet und ausgelegt, daß die Übertragungssteuerungen nacheinander jeweils den Datenverkehr zwischen den ihnen zugeordneten Datenverarbeitungseinheiten und beliebigen anderen Datenverarbeitungseinheiten steuern und nach Beendigung der Datenübertragungen je ein Kommandoübergabesignal über die Sammelleitung an diejenige Übertragungssteuerung senden, die als nächste die Datenübertragung über die Sammelleitung steuern soll. Die von dem ausgesendeten Kommandoübergabesignal adressierte Übertragungssteuerung übernimmt das Kommando für die Datenübertragung auf der Sammelleitung und quittiert die Kommandoübernahme durch Abgabe eines Kommandoübergabesignals an die Sammelleitung.

Bei dem aus der DE-AS 26 31 052 bekannten Datenübertragungssystem erfolgt die Kommandoübertragung auf die jeweils nächste Übertragungssteuerung bzw. die Kommandoübernahme durch die jeweils nächste Übertragungssteuerung unabhängig davon, ob in der das Kommando übernehmenden nächsten Übertragungssteuerung eine Anforderung für einen Datenverkehr über die Sammelleitung vorliegt oder nicht. Auf diese Weise kann wertvolle Übertragungszeit für den Datenverkehr auf der Sammelleitung ungenutzt bleiben oder verlorengehen. Da darüber hinaus das Kommando nur in einer vorbestimmten Reihenfolge von Übertragungssteuerung zu Übertragungssteuerung übergehen werden kann, besteht die Gefahr eines Totalausfalls des Systems, wenn in einer Übertragungssteuerung, auf die das Kommando übertragen werden soll, ein Fehler auftritt. Um diesen Totalausfall des Systems zu vermeiden, sind in den einzelnen Übertragungssteuerungen besondere Überwachungsvorrichtungen vorgesehen.

Aus der GB-PS 11 66 476 ist ein Datenübertragungssystem bekannt, das der bereits weiter oben geschilderten Ringanordnung ähnlich ist und auch die dort genannten Nachteile aufweist. Der Datenübertragungsring wird unter Beachtung eines vorgegebenen Prioritätsschemas benutzt. Die Weitergabe der Übertragungssteuerung auf dem Ring an eine nächste prioritätshohe Einheit ist jedoch ein komplizierter und langwieriger Prozeß.

Der Erfindung liegt die Aufgabe zugrunde, ein Verfahren der gattungsgemäßen Art derart weiterzubilden, daß die Weitergabe der Kontrolle über das Kommunikationsnetzwerk nicht an eine vorbestimmte Reihenfolge von Schnittstellen-Modul zu Schnittstellen-Modul gebunden ist, sondern flexibler gehandhabt wird.

Diese Aufgabe wird durch die kennzeichnenden Merkmale des Patentanspruchs 1 in Verbindung mit den Merkmalen im Oberbegriff gelöst. Die erfindungsgemäße Lösung stellt sicher, daß die Kontrolle über das

Kommunikationsnetzwerk nur an einen solchen Schnittstellen-Modul vergeben wird, der auch tatsächlich die Kontrolle über das Kommunikationsnetzwerk wünscht, und zwar unter Beachtung einer Priorität unter den sendewilligen Schnittstellen-Modulen. Das erfindungsgemäße Verfahren ermöglicht auch ohne weiteres beim Ausfall eines Schnittstellen-Moduls die Weitergabe der Kontrolle über das Kommunikationsnetzwerk an den jeweiligen prioritätshöchsten sendewilligen Schnittstellen-Modul.

Gegenstand der Erfindung ist auch ein Schnittstellen-Modul für ein Kommunikationsnetzwerk, in dem die Nachrichtenübertragung gemäß dem erfindungsgemäßen Verfahren gesteuert wird.

Ein bevorzugtes Anwendungsgebiet der Erfindung sind industrielle Steuerungssysteme, beispielsweise mit Prozeßsteuerungen oder programmierbaren Steuergeräten. Die Erfindung dient insbesondere dazu, derartige Steuerungen in einem Netzwerk miteinander zu verbinden.

Ein bevorzugtes Ausführungsbeispiel der Erfindung soll im folgenden an Hand von Zeichnungen erläutert werden. Es zeigt

Fig. 1 ein Blockschaltbild eines nach der Erfindung gesteuerten Kommunikationsnetzwerks,

Fig. 2 eine bildliche Darstellung eines programmierbaren Steuergerätes, das an das Kommunikationsnetzwerk nach der Fig. 1 angeschlossen ist,

Fig. 3 ein elektrisches Schaltbild eines Schnittstellen-Moduls, der einen Teil des Kommunikationsnetzwerks nach der Fig. 1 bildet,

Fig. 4 ein elektrisches Schaltbild einer Chip-Auswahlschaltung, die einen Teil des Schnittstellen-Moduls nach der Fig. 3 bildet,

Fig. 5 ein elektrisches Schaltbild einer Datenverbindungstreiber- und Empfangsschaltung, die einen Teil des Schnittstellen-Moduls nach der Fig. 3 bildet,

Fig. 6 ein Blockschaltbild des Software-Systems des Schnittstellen-Moduls,

Fig. 7 eine Teildarstellung eines Direktzugriffsspeichers des Schnittstellen-Moduls nach der Fig. 3,

Fig. 8 ein elektrisches Blockschaltbild eines programmierbaren Steuergeräts, das mit dem Schnittstellen-Modul nach der Fig. 3 verbunden ist,

Fig. 9 ein Sprossendiagramm der Kommunikationsbefehle, die in dem programmierbaren Steuergerät nach der Fig. 8 gespeichert sind,

Fig. 10 eine Darstellung eines Privilegien- und Kommunikationssprossenindexabschnitts des Schnittstellen-Modul-Speichers,

Fig. 11 ein Flußdiagramm der Steuergerätabtast- und Nachrichtenausführungsroutine, die ein Teil des Systems nach der Fig. 6 ist,

Fig. 12 ein Flußdiagramm der NETX-Routine, die ein Teil des Systems nach der Fig. 6 ist,

Fig. 13A bis 13D Flußdiagramme der Protokolltreiberroutine, die einen Teil des Systems nach der Fig. 6 bildet, und

Fig. 14 ein Blockschaltbild des Nachrichtendatenflusses im Schnittstellen-Modul nach der Fig. 3.

Beschreibung eines bevorzugten Ausführungsbeispiels der Erfindung

In der Fig. 1 ist ein nach der Erfindung ausgebildetes Kommunikations- oder Übertragungsnetzwerk dargestellt, das eine Datenverbindung mit einem abgeschirmten Kabel 1 aus einem Paar verdrehter Leitungsdrähte 2 und 3 aufweist. Das Kabel, bei dem es sich um ein Produkt der Firma Belden Corporation mit der Nr. 9463 handeln kann, ist in der Lage, mit einem Datenfluß bis zu 57,6 kBd und über eine Entfernung bis zu 3000 m zu arbeiten. Das Kabel 1 ist an seinen Enden mit je einem Widerstand 4 bzw. 5 abgeschlossen und hat in seiner Längsrichtung mehrere voneinander getrennte Zweigleitungen oder Abzweigungen 6. Ein Steuergerät-Schnittstellen-Modul ist an jeden der Abzweige angeschlossen, und auf diese Weise können bis zu 64 getrennte Steuergeräte mit dem Kabel 1 verbunden werden.

Über die zugeordneten Steuergerät-Schnittstellen können an das Kabel 1 verschiedene Arten von Steuergeräten angeschlossen werden. Bei diesen Steuergeräten kann es sich beispielsweise um eine Rechnersteuerung 7 mit Allzweckrechnern handeln, die über eine Schnittstelle 8 für Rechnersteuergeräte an das Kabel 1 angeschlossen sind, oder es kann sich um programmierbare Steuergeräte 9 handeln, die über eine Schnittstelle 10 für programmierbare Steuergeräte mit dem Kabel 1 verbunden sind. Gleichmaßen können Prozeßsteuergeräte 11 über eine Schnittstelle 13 für Prozeßsteuergeräte mit dem Kabel 1 verbunden sein, und es können numerische Steuergeräte oder eine numerische Steuerung 12 über eine Schnittstelle 14 für numerische Steuergeräte an das Kabel 1 angeschlossen sein. Die verschiedenen Arten und die Anzahlen der Steuergeräte hängen von der jeweiligen besonderen Anlage ab. Die Erfindung soll in dieser Hinsicht die erforderliche Flexibilität sicherstellen und die notwendige Zuverlässigkeit gewährleisten.

Die Steuergerät-Schnittstellen 8, 10, 13 und 14 sind als Module ausgebildet, und es handelt sich um Schaltungen, die auf Mikroprozessoren beruhen und so programmiert sind, daß sie die Schnittstelle zwischen dem Netzwerkkommunikationsprotokoll und dem Protokoll des besonderen Steuergerätes bereitstellen, mit dem sie verbunden sind. Die Schaltungen der Steuergerät-Schnittstellen 8, 13 und 14 sind gleich, da die meisten Rechnersteuerungen 7, Prozeßsteuergeräte 11 und numerischen Steuerungen 12 serielle Eingabe/Ausgabe-Kanäle haben, die dem Industriestandard RS232C entsprechen.

Die Schnittstelle 10 für programmierbare Steuergeräte unterscheidet sich von den übrigen Schnittstellen darin, daß sie sich im Eingabe/Ausgabe-Gestell des programmierbaren Steuergerätes befindet und dazu dient, ein Programmierpult 15 mit dem programmierbaren Steuergerät 9 zu verbinden. In dieser Hinsicht ist die Schnittstelle 10 von einer spezielleren Art.

Aus der Fig. 2 geht hervor, daß die Schnittstelle 10 an einem Gestell 16 angebracht ist, das einen Prozessor 17 und Eingabe/Ausgabe-Schnittstellenschaltungen 18 trägt, die im programmierbaren Steuergerät 9 enthalten

sind. Die Steuergerät-Schnittstelle 10 ist so ausgebildet, daß sie in einen der Schlitze paßt, die normalerweise von den Module darstellenden Eingabe/Ausgabe-Schnittstellenschaltungen 18 eingenommen werden, und sie enthält drei Buchsen 19, 20 und 21, die den die Schnittstelle 10 darstellenden Modul über Leitungen 6, 22 und 23 mit dem Kabel 1, dem Programmierpult 15 und dem Prozessor 17 des programmierbaren Steuergeräts verbinden. Die mechanische Konstruktion ist im einzelnen in der US-PS 41 51 580 erläutert. Das Programmierpult 15 ist in der US-PS 40 70 702 beschrieben.

Die Schnittstelle 10 für programmierbare Steuergeräte ist im einzelnen in der Fig. 3 dargestellt. Sie enthält einen 8-Bit-Mikroprozessor 25, der an einen 8-Bit-Datenbus 26 und einen 14-Bit-Adressenbus 27 angeschlossen ist. Es wird ein von der Fa. Zilog, Inc. hergestellter Mikroprozessor Modell Nr. Z-80A verwendet, und der Mikroprozessor 25 wird von einem 3,6864-MHz-Taktgeber 28 angesteuert. Bezüglich des Aufbaus, der Arbeitsweise und des Befehlssatzes dieses Mikroprozessors wird auf eine Druckschrift "Z80-CPU Technical Manual" herausgegeben von Zilog, Inc., 1976, verwiesen.

Der Mikroprozessor 25 enthält eine Reihe von Steueranschlüssen IORQ, M1, RD, WR und MREQ, die entsprechend zugeordnete und mit anderen Elementen der Schaltungsanordnung verbundene Steuerleitungen ansteuern. Weiterhin sind am Mikroprozessor 25 zwei Unterbrechungsanschlüsse NMI und INT vorgesehen, die über zugehörige Steuerleitungen von verschiedenen Schaltungselementen angesteuert werden.

Die Arbeitsweise des Mikroprozessors 25 und damit die Arbeitsweise der gesamten Schnittstellenschaltungsanordnung erfolgt unter der Steuerung von Programmen, die in einem Nur-Lese- oder Festwertspeicher 29 gespeichert sind. Der Festwertspeicher enthält zwei durch UV-Licht löschbare 4 k · 8-PROM-Schaltungen, von denen jede über Anschlüsse 30 und 31 durch eine Chip-Auswahlschaltung 32 getrennt freigegeben werden kann. Ein 8-Bit-Byte an Daten wird aus einer adressierten Zeile des Festwertspeichers 29 gelesen, wenn von einem ODER-Glied 34 ein niedriger Signalpegel an einen OE-Anschluß 33 gelegt wird. Die ausgelesenen Daten erscheinen am Datenbus 26 und gelangen in den Mikroprozessor 25, der dann die angegebene Operation ausführt.

Daten einschließlich von Nachrichten, die vom Kabel 1 empfangen werden, und Nachrichten, die an das Kabel 1 abgegeben werden, werden in einem Speicher mit wahlfreiem Zugriff oder einem Direktzugriffsspeicher 35 gespeichert. Der Direktzugriffsspeicher 35 enthält zwei statische 1k · 8-RAM-Schaltungen, von denen jede getrennt über Anschlüsse 36 und 37 durch die Chip-Auswahlschaltung 32 freigegeben werden kann. Ein 8-Bit-Byte an Daten wird in den Direktzugriffsspeicher 35 eingeschrieben oder aus ihm ausgelesen, wenn Steuersignale an einen OE-Anschluß 38 und einen WE-Anschluß 39 durch das ODER-Glied 34 bzw. ein ODER-Glied 40 angelegt werden.

Über eine serielle Eingabe/Ausgabe-Schaltung 41, die auch mit SIO bezeichnet wird, gelangt Information vom Datenbus 26 zu der Zweigleitung 6 und der Leitung 23 sowie von diesen Leitungen zum Datenbus. Die eine serielle Eingabe/Ausgabe darstellende Schaltung 41 ist ein handelsüblicher peripherer Dualkanal-Mehrfunktionsbaustein, der an 8-Bit-Datenbytes eines Serien/Parallel- bzw. Parallel/Serien-Umsetzung vornimmt. Der SIO 41 spricht auf Kommandodaten am Datenbus 26 an, um an jedem seiner beiden Kanäle eine Reihe von Funktionen auszuführen, wobei durch diese Kommandos, die unter der Steuerung von im Festwertspeicher 29 gespeicherten Programm erzeugt werden, die Arbeitsweise jedes seriellen Eingabe/Ausgabe-Kanals den verschiedenen Übertragungs- oder Kommunikationsprotokollen an den Leitungen 6 und 23 angepaßt wird.

Der SIO 41 wird über die Mikroprozessor-Steuerleitungen IORQ, M1 und RD sowie über die Systemtaktleitung angesteuert. Eine Leitung A3 im Adressenbus 27 gibt den SIO 41 frei, und Leitungen A0 und A1 wählen Kanäle A oder B aus und zeigen an, ob es sich bei der Information am Datenbus 26 um ein Kommando oder um Daten handelt. Ein Sende- und Empfangsdatentaktanschluß für den Kanal B wird vom Q-Ausgang eines D-Flipflop 43 angesteuert, das seinerseits von einer Zählerzeitgeberschaltung 44 angesteuert wird, die auch mit CTC bezeichnet ist. Der SIO 41 ist auch zur Ansteuerung mit dem Mikroprozessor-Unterbrechungsanschlüssen INT und NMI verbunden, und ist über eine Unterbrechungsfreigabeausgangsleitung 45 mit der Bezeichnung IEO an einen Unterbrechungsfreigabeeingang IEI der Schaltung 44 angeschlossen. Die Anschlüsse des A-Kanals und des B-Kanals des SIO 41 sind mit einer Datenverbindungs-Treiber/Empfänger-Schaltung 46 bzw. einer Steuergerät-Treiber/Empfänger-Schaltung 47 verbunden.

Die Arbeitsweise und der Aufbau des Bausteines SIO bzw. der Schaltung 41 ist in einer Druckschrift "Z-80 SIO Technical Manual", herausgegeben von Zilog, Inc., 1977, beschrieben. Der SIO 41 wird hier benutzt, um über den Kanal B eine Vollduplex-Asynchron-Serien-Verbindung mit dem Prozessor des programmierbaren Steuergeräts herzustellen und um über den Kanal A eine Halbduplex-Synchron-Serien-Verbindung mit anderen Schnittstellenmodulen des Netzwerks herzustellen. Der Kanal B hat Datenflüsse von 9600 und 19 200 Bd, und der Kanal A hat einen Datenfluß von 57 600 Bd.

Die Auswahl der Baudraten, der Stationsadresse und wahlfreier Maßnahmen oder Zusatzmaßnahmen erfolgt über zwei Gruppen von Schaltern 48 und 49, die über einen Eingabemultiplexer 50 mit dem Datenbus 26 verbunden sind. Der Eingabemultiplexer 50 enthält zwei 4-Bit-Multiplexerschaltungen, deren Auswahlanschluß SEL gemeinhin mit der Leitung A0 des Adressenbusses 27 und deren Freigabeanschluß G über eine Leitung 51 mit der Chip-Auswahlschaltung 32 verbunden ist. Die Schalter 48 enthalten DIP-Schalter, die eine 8-Bit-Stationsadresse an die A-Eingänge des Multiplexers 50 liefern. Diese Schalter 48 werden von Hand auf eine Stationsadresse von 0 bis 254 eingestellt. Die Schalter 49 sind mit den B-Eingängen des Multiplexers 50 verbunden und enthalten einen DIP-Schalter, der zur Auswahl der Baudrate dient, sowie inpolige Ein- und Ausschalter, die zur Auswahl von wahlweisen oder zusätzlichen Maßnahmen dienen.

Die Kommunikation durch die Leitung 22 mit dem Programmierpult 15 erfolgt über einen Universal-Synchron/Asynchron-Empfänger/Sender 55, der im folgenden mit USART bezeichnet wird. Der USART 55 ist eine handelsübliche integrierte Schaltung, die durch eine an die Chip-Auswahlschaltung 32 angeschlossene Leitung 56 freigegeben wird. Der USART 55 wird über eine Leitung 57 getaktet, die an den Systemtaktgeber 28

angeschlossen ist und auch zur Zählerzeitgeberschaltung 44 führt, die einen mit CTC bezeichneten Baustein darstellt. Die Auswahl von Kommandodaten oder Eingabe/Ausgabe-Daten erfolgt durch die mit einem C/D-Anschluß 58 verbundene Leitung A0 im Adressenbus 27, und der USART 55 wird über die Mikroprozessor-Steuerleitungen WR und RD betrieben. Der USART 55 erhält ein Baudraten-Taktsignal vom CTC 44 über ein Umkehr-Glied 59, und der USART 55 erzeugt einen Unterbrechungsaufwurf für den Mikroprozessor 25 über eine RxRDY-Leitung 60, ein ODER-Glied 61 und ein NAND-Glied 62, wenn Daten vom Programmierpult 15 empfangen werden.

Der USART 55 stellt mit einer auswählbaren Baudrate von 9600 oder 19 200 Bd eine Vollduplex-Asynchron-Serien-Verbindung mit dem Programmierpult 15 her. Die Arbeitsweise des USART 55 erfolgt unter der Steuerung von Programmen, die im Festwertspeicher 29 gespeichert sind und die auch den SIO 41 betreiben, um die Datenverbindung zwischen dem Programmierpult 15 und dem Prozessor 17 des im folgenden auch mit PC bezeichneten programmierbaren Steuergeräts herzustellen.

Die Baudrate, mit der der SIO 41 und der USART 55 arbeiten, wird vom CTC 44 gesteuert. Der CTC 44 ist eine handelsübliche integrierte Schaltung, die mit dem Datenbus 26 verbunden ist, um Kommandos zu empfangen, die die Baudraten für die drei seriellen Eingabe/Ausgabe-Kanäle erstellen. Der CTC 44 wird von den Adressenbusleitungen A0, A1 und A5 freigegeben, die mit entsprechenden Anschlüssen CS0, CS1 und CE verbunden sind. Der CTC 44 wird über die Mikroprozessor-Steuerleitungen IORQ, RD und M1 betrieben. Der CTC 44 erhält das 3,6864-MHz-Taktsignal vom Taktgeber 28 als auch 1,8432-MHz-Triggersignal über die Leitung 57. Der CTC 44 kann ein Unterbrechungssignal für den Mikroprozessor-Anschluß INT erzeugen, und er erzeugt drei Taktsignale, nämlich ein Taktsignal für den USART 55 über das Umkehr-Glied 59, ein zweites Taktsignal für das D-Flipflop 43 über eine Leitung 65 und ein drittes Taktsignal für die Datenverbindungs-Treiber/Empfänger-Schaltung 46 über eine Leitung 66. Eine vollständige Beschreibung des Aufbaus und der Wirkungsweise des CTC 44 findet man in einer Druckschrift "Z80-CTC Technical Manual", herausgegeben von der Firma Zilog, Inc., 1977.

Aus den Fig. 2 und 3 geht hervor, daß fünf Indikatoren oder Anzeigelampen 68 über einen Ausgabe-Kanal 69 mit dem Adressenbus verbunden sind und oberhalb der Buchse 19 am Rand des Moduls der Schnittstelle 10 montiert sind. Der Ausgabekanal 69 stellt eine oktad-adressierbare Verriegelung dar, bei der es sich um eine handelsübliche integrierte Schaltung handelt. Diese Schaltung wird von der Chip-Auswahlschaltung über eine Leitung 70 freigegeben. Ferner weist sie vier Eingänge auf, die mit den Leitungen A0, A1, A2 und A4 des Adressenbusses 27 verbunden sind. Fünf Ausgangsanschlüsse des Ausgabekanals 69 sind mit den Anzeigelampen 68 verbunden. Ein weiterer Ausgang des Ausgabekanals 69 führt zu dem ODER-Glied 61. Ein siebter Ausgangsanschluß ist über eine Trägererfassungsrücksetzleitung 71 mit der Datenverbindungs-Treiber/Empfänger-Schaltung 46 verbunden. Der 3-Bit-Code an den Adressenbusleitungen A0, A1 und A2 wählt einen der acht Ausgangsanschlüsse aus, und der Logikzustand der Adressenbusleitungen A4 wird durch die Chip-Auswahlschaltung 32 in den Ausgabekanal 69 getaktet. Auf diese Weise kann irgendeine der Anzeigelampen 68 ein- oder ausgeschaltet werden, und eine "Software-Unterbrechung" kann über das ODER-Glied 61 erzeugt werden.

Die Fig. 4 läßt erkennen, daß die Chip-Auswahlschaltung 32 in erster Linie zwei 2-Bit-auf-4-Bit-Decodierschaltungen 73 und 74 enthält. Die beiden Eingänge der Decodierschaltung 73 sind mit den Adressenbusleitungen A10 und A13 verbunden, wohingegen die vier Ausgänge an die Festwertspeicher-Chip-Auswahlanschlüsse 30 und 31 sowie an die Direktzugriffsspeicher-Chip-Auswahlanschlüsse 36 und 37 angeschlossen sind. Die Eingänge an der zweiten Decodierschaltung 74 sind mit den Adressenbusleitungen A6 und A7 verbunden, und ihr Ausgabefreigabeanschluß OE wird über Tore oder Verknüpfungsglieder 75 und 76 von Signalen an den Steuerleitungen IORQ und M1 angesteuert. Drei Ausgänge der zweiten Decodierschaltung 74 sind mit den Leitungen 51, 56 und 70 verbunden, über die in Abhängigkeit vom Zustand der Adressenbusleitungen A6 und A7 der Eingabemultiplexer 50, der USART 55 und der Ausgabekanal 69 freigegeben werden.

Im folgenden wird insbesondere auf die Fig. 3 und 5 Bezug genommen. Die Datenverbindungs-Treiber/Empfänger-Schaltung 46 verbindet die Zweigleitung 6 mit dem Kanal A des SIO 41. Der SIO 41 gibt Daten über eine Leitung 85 ab, die mit einem Eingang eines exklusiven ODER-Glieds 86 verbunden ist. Der andere Eingang des exklusiven ODER-Glieds 86 wird von einem Sendetaktsignal angesteuert, das von einem 4-Bit-Binärlähler 87 erzeugt wird. Am Ausgang des exklusiven ODER-Glieds 86 treten zweiphasig codierte Daten nach dem "Manchester-Code" auf. Der Binärlähler 87 wird über die Leitung 66 angesteuert, die zum CTC 44 führt.

Auf diese Weise wird die Geschwindigkeit der Datenübertragung von Programmbefehlen gesteuert, die während der Systeminitialisierung ausgeführt werden.

Die zweiphasig codierten Ausgangsdaten werden von einem zweiten exklusiven ODER-Glied 88 invertiert, und die nicht invertierten Daten als auch die invertierten Daten werden den Eingängen von UND-Gliedern 89 und 90 von einem freigegebenen UND-Glied 91 angesteuert. Ein Eingang und UND-Glieds 91 ist mit dem Sendeaufwurf-Ausgangsanschluß RTSA des SIO 41 verbunden. Die Ausgänge der UND-Glieder 89 und 90 sind an die Primärwicklung eines Trenntransformators 92 angeschlossen. Der Transformator 92 hat ein Windungsverhältnis von 1,2 bis 4,0, um eine Spannungsaufwärtstransformation vorzusehen, bevor das zweiphasig codierte Signal an das Kabel 1 gelegt wird. Sowohl die Primärwicklung als auch die Sekundärwicklung des Transformators weisen eine Mittenanzapfung auf. Wenn die UND-Glieder 89 und 90 vom SIO 41 freigegeben sind, wird somit ein 57,6-kHz-Trägersignal mit einer Differentialspannung von etwa 6 V an das Kabel 1 gelegt. Wenn über die Leitung 85 Daten ausgegeben werden, wird dieses Trägersignal moduliert, um an andere Stationen, die mit dem Kabel 1 verbunden sind, zweiphasig codierte Nachrichten auszusenden.

Wie man ferner der Fig. 5 entnehmen kann, werden von der Steuergerät-Schnittstelle 10 Nachrichten über das Kabel 1 empfangen und an die Primärwicklung eines zweiten Trenntransformators 93 gelegt. Der Transformator 93 hat ein Windungsverhältnis von 6:15, und beide Wicklungen weisen eine Mittenanzapfung auf. Die Sekundärwicklung des Transformators 93 ist über Widerstände 94 und 95 an die Eingänge eines Differential- oder Differenzenverstärkers 96 angeschlossen. Zwei Zenerdioden 97 und 98 dienen als Schutz gegenüber einer

übermäßig hohen Eingangsspannung, und der Verstärker 96 arbeitet wie ein Leitungsempfänger, um das empfangene Signal in TTL-Spannungspegel umzuformen (TTL = Transistor-Transistor-Logik). Das Ausgangssignal des Verstärkers 96 wird über eine Leitung 97 dem Datenempfangseingangsanschluß RXDA des SIO 41 zugeführt und darüber hinaus an ein Paar von exklusiven ODER-Gliedern 98 und 99 gelegt. Die exklusiven ODER-Glieder 98 und 99 arbeiten in Kombination mit einem Kondensator 100, um immer dann einen positiven Spannungsimpuls zu erzeugen, wenn das empfangene Datensignal seinen Logikzustand ändert. Diese Spannungsimpulse werden über ein UND-Glied 102 und ein Umkehr-Glied 103 dem Verriegelungs- oder Einklinkeingang eines 4-Bit-Binärzählers 104 zugeführt. Der Zähler 104 und die zugehörige Schaltung dienen zum Decodieren der zweiphasig codierten Daten. Der Zähler 104 wird über die Leitung 66 getaktet, die vom CTC 44 angesteuert wird. Der Zähler 104 ist voreingestellt (auf einen Zählwert von 2) und zählt vorwärts bis zu einem Zählwert von vier. Bei diesem Zählwert erzeugt der Zähler an einer Leitung 105 einen Impuls, der zu einem Empfangstakteingangsanschluß RXCA am SIO 41 gelangt. Nachdem ein Zählwert von acht erreicht ist, nimmt der Q8-Ausgang des Zählers 104 einen hohen Pegel an, und das UND-Glied 102 wird über eine Leitung 106 freigegeben, um den nächsten positiven Spannungsimpuls zu empfangen. Nach Empfang dieses Spannungsimpulses wird der Zähler 104 erneut auf den Zählwert von zwei eingestellt, und der Zyklus wiederholt sich.

Weiterhin kann man den Fig. 3 und 5 entnehmen, daß der Ausgang des als Leitungsempfänger arbeitenden Verstärkers 96 auch mit dem Eingang eines ersten monostabilen Multivibrators 108 verbunden ist, dessen Q-Ausgang einen zweiten monostabilen Multivibrator 109 und ein UND-Glied 110 ansteuert. Der Q-Ausgang des monostabilen Multivibrators 109 ist mit einem zweiten Eingang des UND-Glieds 110 verbunden, und die Rücksetzanschlüsse beider Multivibratoren 108 und 109 sind an die Trägererfassungsrücksetzleitung 71 angeschlossen, die vom Ausgabekanal 69 angesteuert wird. Der Ausgang des UND-Glieds 110 ist mit dem Datenträgererfassungsanschluß DCDA des SIO 41 über eine Trägererfassungsleitung 111 verbunden.

Wie noch im einzelnen beschrieben wird, erzeugt der SIO 41 eine Unterbrechung, und zwar sowohl beim Erscheinen des Trägersignals am Kabel 1 als auch beim Verschwinden des Trägersignals. Diese Ereignisse werden herangezogen, um zu bewerkstelligen, daß andere Stationen mit dem Kabel 1 verbunden sind und aktiv sind. Durch Überprüfen des Zustands der Leitung 111 zu einer geeigneten Zeit, kann das System feststellen, ob eine andere Station auf ein Abfrage- oder Aufruf-Kommando geantwortet hat.

Aus der obigen Erläuterung geht hervor, daß die PC-Schnittstelle 10 (PC = programmierbares Steuergerät) eine Anzahl von verschiedenen Aufgaben ausführen muß. Sie muß vom Kabel 1 Nachrichten empfangen und an das programmierbare Steuergerät 9 weiterleiten können. Sie muß vom programmierbaren Steuergerät 9 Nachrichten empfangen und an das Kabel 1 zu anderen Stationen weiterleiten können. Sie muß Daten zwischen dem programmierbaren Steuergerät 9 und seinem Programmladegerät oder Programmierpult 15 austauschen können. All diese Aufgaben werden von dem Mikroprozessor 25 unter der Leitung von Befehlen ausgeführt, die in einer Reihe von Programmen enthalten sind. Diese Programme sind im Festwertspeicher 29 gespeichert, und sie werden vom Mikroprozessor 25 in Sequenz ausgelesen und ausgeführt.

Im folgenden soll auch auf die Fig. 6 Bezug genommen werden. Die Programme, die den Mikroprozessor 25 durch die Aufgaben leiten und führen, die er ausführen muß, sind in Modulen angeordnet. Jedes Programm enthält eine oder mehrere Routinen oder Subroutinen. Es gibt zwei Arten von Routinen, und zwar solche, die aufgrund von Ereignissen ausgeführt werden, die Unterbrechungen erzeugen, und solche, die aufgrund von Ereignissen ausgeführt werden, die als Ergebnis der Ausführung von Programmen auftreten.

Die erste Art von Routinen wird mit Treiber bezeichnet, und das System enthält das folgende: eine Datenverbindungstreiberoutine 115 und eine Protokolltreiberoutine 116, die aufgrund von Unterbrechungen ausgeführt werden, die durch den Kanal A des SIO 41 erzeugt werden, eine Steuergerätreiberoutine 117, die aufgrund einer Unterbrechung ausgeführt wird, die durch den Kanal B des SIO 41 erzeugt wird, eine Programmierpulttreiberoutine 118, die aufgrund einer Unterbrechung ausgeführt wird, die durch den USART 55 erzeugt wird, und eine 25-ms-Takttreiberoutine 119, die aufgrund einer Unterbrechung ausgeführt wird, die durch den CTC 44 erzeugt wird.

Die zweite Art von Routinen in dem System wird mit "Task" bezeichnet. Dazu gehören eine Programmierpult-handhabungsroutine 120, eine Antwortroutine 121, eine Netzwerkkommandoausführungsroutine 122, eine Steuergerätabtast- und Nachrichteneinleitroutine 123 sowie eine Zeitgeberoutine 124. Diese Tasks werden beim Hoch- oder Anfahren eingeleitet und haben kein Ende. Eine Task kann sich selbst aufrechterhalten und auf das Auftreten eines Ereignisses warten, bevor sie fortfährt, oder sie kann unterbrochen werden, und nach Bedienung der Unterbrechung kann man wieder in sie eintreten.

Da zur gleichen Zeit mehr als eine Unterbrechung auftreten und mehr als eine Task im Zustand für die Ausführung sein können, wird ein Hardware- und Software-Prioritätssystem benutzt, um die Mikroprozessorausführungszeit optimal zuzuteilen. Wie es aus der Fig. 3 hervorgeht, ist der Anschluß INT am Mikroprozessor 25 mit dem CTC 44, dem USART 55 und dem SIO 41 verbunden, und zwar derart, daß der Kanal A des SIO, der Kanal B des SIO, der USART und der CTC in der genannten Reihenfolge Priorität haben. Fordern beispielsweise der CTC 44 und der Kanal B des SIO eine Unterbrechung an, wird der Kanal B des SIO zuerst von der Steuergerätreiberoutine 117 bedient, und anschließend wird die Unterbrechungsanforderung durch den CTC 44 von der Takttreiberoutine 119 bedient. Die Treiberoutinen 115 bis 119 werden somit von der Unterbrechungsschaltung prioritätsmäßig verarbeitet.

Im folgenden wird wieder die Fig. 6 herangezogen. Wenn keine Unterbrechungen bedient werden, kann eine Task ausgeführt werden, und es ist die Aufgabe oder Funktion einer mit Scheduler bezeichneten Routine 125, diejenige Task zu bestimmen, die die höchste Priorität hat. In den Scheduler 125 wird eingetreten, nachdem eine Start- und Initialisieroutine 126 ausgeführt ist, und von diesem Zeitpunkt an wird in den Scheduler aufgrund einer durch Software eingeleiteten Unterbrechung eingetreten. Die Fig. 3 läßt erkennen, daß diese Unterbrechung durch den Ausgabekanal 69 erzeugt wird und über das ODER-Glied 61 und das NAND-Glied 62 an den

Anschluß INT des Mikroprozessors 25 gelegt wird. Diese Unterbrechung hat die niedrigste Priorität. Anders als die anderen Unterbrechungen, die von Ereignissen resultieren, welche außerhalb der Steuergerät-Schnittstelle 10 auftreten, erscheint diese Software-Unterbrechung, wenn vom Mikroprozessor ein "Unterbrechungsbefehl" ausgeführt wird, um an der niedrigstwertigen Bitstelle des Ausgabekanals 69 eine "1" auszugeben. Es ist somit für irgendein Programm im System möglich, den Scheduler 125 durch Ausführung dieses Unterbrechungsbefehls zu rufen. Es sei allerdings bemerkt, daß dies eine asynchrone durch Software eingeleitete Unterbrechung ist, die zurückgestellt wird, bis die anderen Unterbrechungsrouinen inaktiv sind.

Zum folgenden wird insbesondere auf die Fig. 6 und 7 verwiesen. Der Scheduler 125 bestimmt durch Bezugnahme auf eine Tasksteuerblockprioritätstabelle 127, welche Task die höchste Priorität hat. Die Tabelle 127 ist im Direktzugriffsspeicher 35 gespeichert und sie enthält für jede der fünf Tasks 120 bis 124 in dem System ein 8-Bit-Flaggenwort, eine 8-Bit-Maske und einen Stapelzeiger. Die Reihenfolge, in der diese Daten in der Tabelle 127 angeordnet sind, bestimmt die prioritätshöchste Task, und der Scheduler 125 ist wirksam, um jeden Tasksteuerblock zu überprüfen, und zwar zwecks Bestimmung der prioritätshöchsten Task, die zum Durchlauf bereit ist. Die Bits 1 bis 7 des Flaggenwortes in jedem Tasksteuerblock werden als Ereignisflaggen benutzt, und das Bit 0 ist die Durchlaufflagge. Der Scheduler 125 nimmt eine logische UND-Verknüpfung des Flaggenwortes und der Maske vor, und wenn das Ergebnis ungleich null ist, ist die Task zur Ausführung bereit.

Der Scheduler 125 leitet die Ausführung einer Task durch Verwendung des Stapelzeigers ein, um den "Zusammenhang" der Task in den Mikroprozessor 25 zurückzustellen. Der Zusammenhang einer Task ist der Mikroprozessorzustand, der durch die Inhalte der Mikroprozessorregister AF, BC, DE, HL, IX, SP und PC dargestellt ist. Wenn eine Task gesperrt oder unterbrochen wird, wird ihr Zusammenhang dadurch aufbewahrt, daß der Inhalt dieser Register auf den Stapelspeicher geschoben wird und daß der Stapelzeiger im Tasksteuerblock gespeichert wird. Der Scheduler 125 startet erneut die Task, indem ihr Zusammenhang vom Stapelspeicher "gepufft" wird und der nächste Taskbefehl ausgeführt wird, der vom Programmzähler angezeigt wird.

Wenn die Task bis zu ihrer Vervollständigung durchgelaufen ist, d. h. typischerweise bis eine Eingabe/Ausgabe-Operation eingeleitet wird, ruft die Task eine Subroutine WAITFER, die das Durchlaufbit und die Maskenbits im Tasksteuerblock modifiziert und durch Ausführung des "Unterbrechungsbefehls" eine Software-Unterbrechung einleitet. Der Zusammenhang der Task wird auf den Stapelspeicher gestoßen, und in den Scheduler 125 wird wieder eingetreten. Der Scheduler wird ausgeführt, um die nächste Task zu bestimmen, die zu durchlaufen ist. Bevor sich eine Task in dieser Weise selbst zur Ruhe begibt, erstellt die Daten in einer Warteschlange oder in einem Puffer und setzt dann eine Ereignisflagge in einer der anderen Tasks in dem System. Wenn der Scheduler 125 sequentiell durch die Tasksteuerblockprioritätstabelle 127 läuft, entdeckt er die Ereignisflagge, die von der zuvor ausgeführten Task gesetzt worden ist, und aktiviert die neue Task, um die Daten in der Warteschlange oder im Puffer zu bearbeiten. Eine Auflistung der Scheduleroutine 125 befindet sich im Anhang A.

Es wird wieder auf die Fig. 3 und 6 verwiesen. Ein Beispiel für die Art und Weise, in der die Treiber und Tasks in Kombination mit dem Scheduler arbeiten, ist durch die Takttreiberoutine 119 und die Zeitgebertask 124 vorgesehen. In die Takttreiberoutine 119 wird immer dann eingetreten, wenn der 25-ms-Taktgeber im CTC 44 abgelaufen ist. Dieses Ereignis erzeugt eine Hardware-Unterbrechung, die das System auf die Takttreiberoutine 119 einweist. Die Treiberoutine 119 setzt die Ereignisflagge in dem Tasksteuerblock für die Zeitgebertroutine 124, um anzuzeigen, daß ein 25-ms-Tick des Taktgebers aufgetreten ist. Die Treiberoutine erzeugt dann eine Software-Unterbrechung, indem sie den Unterbrechungsbefehl ausführt. Als ein Ergebnis wird in den Scheduler 125 eingetreten, und der Scheduler prüft die Tasksteuerblockprioritätstabelle 127, um festzustellen, ob irgendeine Task zum Durchlauf bereits oder fertig ist. Die Ereignisflagge für die Zeitgebertroutine 124 wird gesetzt, und unter der Annahme, daß keine prioritätshöheren Tasks zum Durchlaufen fertig sind, zweigt der Scheduler 125 zur Zeitgebertroutine 124 ab. Die Zeitgebertroutine 124 speichert Zählschritte, die jeweils nach dem Auftreten eines 25-ms-Ticks um eins dekrementiert werden. Ein beispielsweise auf zwei voreingestellter Zähler ist nach dem Auftreten von zwei Ticks oder 50 ms auf null dekrementiert. Wenn dies auftritt, setzt die Zeitgebertroutine 124 die Ereignisflagge im Tasksteuerblock für die Steuergerätabtast- und Nachrichteneinleitroutine 123. Nachdem all ihre Zähler dekrementiert sind, erzeugt die Zeitgebertroutine die Software-Unterbrechung, und es wird in den Scheduler 125 eingetreten. Der Scheduler erfaßt, daß die Ereignisflagge für die Routine 123 gesetzt worden ist, und auf dies Weise wird die Routine 123 alle 50 ms ausgeführt, um irgendwelche Nachrichten vom programmierbaren Steuergerät zu lesen, die fertig sind, um zu einer anderen Station des Übertragungs- oder Kommunikationsnetzwerks gesendet zu werden.

Um die Art und Weise zu verstehen, in der die Software-Module arbeiten und zusammenwirken, um Nachrichten vom Kabel 1 zu empfangen und an das Kabel 1 abzugeben, soll zunächst das programmierbare Steuergerät 9 kurz erläutert werden. Eine ausführliche Erläuterung dieses programmierbaren Steuergeräts findet man in der US-PS 41.65.534.

Im folgenden wird insbesondere auf die Fig. 8 Bezug genommen. Das programmierbare Steuergerät 9 enthält einen Universal-Asynchron-Empfänger/Sender UAR/T 200, der über ein Kabel 201 an den Kanal B des SIO der Steuergerät-Schnittstelle 10 angeschlossen ist. Der UAR/T 200 steht mit einem Steuergerät-Adressenbus 202 in Verbindung und ist über UAR/T-Datentore oder -Gatter 204 an einen 8-Bit-Datenbus 203 angeschlossen. Der Adressenbus 202 und der Datenbus 203 sind mit einem Steuergerätprozessor 205 verbunden, der den Datenfluß auf dem Datenbus 203 steuert, indem er Adressen auf dem Adressenbus 202 erzeugt, die die Quelle und den Bestimmungsort der Daten festlegen. Eine Datenquelle oder ein Datenbestimmungsort in der UAR/T 200. Wenn ein 8-Bit-Byte an Daten über den UAR/T 200 von der Steuergerät-Schnittstelle 10 empfangen wird, unterbricht der UAR/T 200 die Arbeitsweise des Prozessors 205, und eine UAR/T-Serviceroutine wird ausgeführt, um das Datenbyte durch den Datenbus 203 einzugeben. Andererseits kann der Steuergerätprozessor 205 aufgrund eines Kommandos von der Steuergerät-Schnittstelle 10 ein 8-Bit-Byte an Daten in den UAR/T 200 schreiben und seine

serielle Übertragung durch das Kabel 201 zur Steuergerät-Schnittstelle 10 einleiten.

Der Steuergerät-Adressenbus 202 und der Steuergerät-Datenbus 203 sind auch über Eingabe/Ausgabe-Adressentore oder -Adressengatter 201 bzw. über Eingabe/Ausgabe-Datentore oder -Datengatter 211 mit Eingabe/Ausgabe-Schnittstellenschaltungen 206 bis 209 verbunden. Die E/A-Schnittstellenschaltungen 206 bis 209 sind mit Fühlgeräten verbunden, beispielsweise mit Schaltern, und sind mit Stellgeräten verbunden, beispielsweise mit Motoren und Hubmagneten angesteuerten Maschinen. Der Steuergerät-Prozessor 205 führt periodisch eine Eingabe/Ausgabe-Abtastroutine aus, die Daten von den E/A-Schnittstellenschaltungen 206 bis 209 eingibt und Daten an die E/A-Schnittstellenschaltungen 206 bis 209 abgibt, um die Arbeitsweise der Maschinen zu steuern. Dieser Vorgang tritt typischerweise alle 20 ms oder in noch kürzeren Zeitabständen auf.

Der Steuergerät-Prozessor 205 ist eine programmierte Maschine, und die Programme, die seine Arbeitsweise steuern, einschließlich der UAR/T-Bedienungsroutine und der E/A-Abtastroutine, sind in einem Festwertspeicher gespeichert, der in den Zeichnungen nicht dargestellt ist. Andererseits sind das Steuerprogramm, das bestimmt, wie die gesteuerte Maschine betrieben werden soll, und andere Daten, die zum Ausführen dieses Betriebs verwendet werden, in einem Direktzugriffsspeicher 212 gespeichert, der sowohl mit dem Adressenbus 202 als auch mit dem Datenbus 203 verbunden ist. Der Steuergerät-Prozessor 205 arbeitet derart, daß er die Steuerprogrammbeefehle sequentiell aus dem Speicher 212 liest und die Funktionen ausführt, die durch einen Operationscode in jedem Steuergerätfbefehl angegeben sind. Diese Befehle umfassen solche, die zum Prüfen des Zustands eines ausgewählten Fühlgerätes dienen, das mit einer der E/A-Schnittstellenschaltungen 206 bis 209 verbunden ist, und solche, die zum Betätigen der Stellgeräte an den gesteuerten Maschinen dienen. Zum Ausführen einer Vielzahl von Steuerfunktionen sind zahlreiche verschiedene Operationscodes verfügbar, und bezüglich einer detaillierten Erläuterung eines Steuerbefehlssatzes wird auf die US-PS 41 65 534 verwiesen.

Ein Hauptziel der Erfindung besteht darin, daß programmierbare Steuergerät 9 zu befähigen, Nachrichten mit Kommandos oder Daten an andere Stationen im Kommunikations- oder Übertragungsnetzwerk abzugeben. Ferner ist in Betracht gezogen, daß andere Stationen an das programmierbare Steuergerät 9 Nachrichten aussenden können. Das programmierbare Steuergerät 9 kann beispielsweise so programmiert sein, daß es eine Nachricht an das Rechnersteuersystem oder die Rechnersteuerung 7 aussendet, um anzuzeigen, daß auf einer Fertigungsstraße ein Bauteil gerade einen bestimmten Punkt passiert hat. Diese Nachricht soll immer dann abgegeben oder ausgesendet werden, wenn ein besonderer Grenzscharter schließt, der mit den E/A-Schnittstellenschaltungen 206 bis 209 verbunden ist. Folglich ist in dem Steuerprogramm ein Befehl enthalten, der festlegt, wenn dieses Ereignis auftritt. Das Aussenden einer Nachricht durch das programmierbare Steuergerät 9 erfolgt somit als Ergebnis der Steuerprogrammausführung.

Umgekehrt können andere Steuergeräte im Kommunikations- oder Übertragungsnetzwerk an das programmierbare Steuergerät 9 Nachrichten aussenden, die ein besonderes Stellgerät betreffen, das mit den E/A-Schnittstellenschaltungen 206 bis 209 verbunden ist und betätigt werden soll. Solche Daten werden über den UAR/T 200 in den Steuergerät-Prozessor 205 eingegeben und dann in einen bezeichneten Platz des Direktzugriffsspeichers 212 geschrieben.

Im folgenden wird insbesondere auf die Fig. 8 und 9 Bezug genommen. Um Nachrichten für andere Stationen zu erzeugen und um andere Stationen in die Lage zu versetzen, Daten in den Direktzugriffsspeicher 212 des programmierbaren Steuergeräts zu schreiben, enthält das Steuerprogramm eine oder mehrere Übertragungs- oder Kommunikationssprossen 213. Die Kommunikationssprossen 213 enthalten Steuergerätfbefehle, die von der Steuergerät-Schnittstelle 10 benutzt werden, um zu bestimmen, wo auszusendende Nachrichtendaten im Direktzugriffsspeicher 212 des Steuergerätes gespeichert werden sollen und an welche Station die Nachrichtendaten auszusenden sind. Diese Kommunikationssprossen 213 ermöglichen es auch der Steuergerät-Schnittstelle 10 zu bestimmen, ob andere Stationen Daten aus dem Direktzugriffsspeicher 212 lesen oder in diesen Speicher einschreiben können, und wenn dies zutrifft, um welche Speicherplätze es sich handelt. Im Anhang H ist das Format der Kommunikationssprossen 213 unter Verwendung von Befehlen nach Art des programmierbaren Steuergeräts dargestellt. Die Fig. 9 zeigt für diejenigen, mit der Technik programmierbarer Steuergeräte vertraut sind, die entsprechenden Sprossendiagramme.

Die Steuergerätfbefehle, die zur Bildung der Kommunikationssprossen 213 verwendet werden, üben nicht ihre gewöhnlichen Steuerfunktionen aus, sondern dienen vielmehr als Codes, die die Schnittstelle 10 des programmierbaren Steuergeräts identifizieren kann, wenn sie die Kommunikationssprossendaten aus dem Steuergerät-Direktzugriffsspeicher 212 ausliest. Viele dieser Daten sind zur Verwendung durch die Tasks 120 bis 124 im Steuergerät-Schnittstellen-Direktzugriffsspeicher 35 gespeichert. Obgleich bei dem erläuterten bevorzugten Ausführungsbeispiel existierende Operationscodes des programmierbaren Steuergeräts verwendet werden, können auch neue Steuergerätfbefehle definiert werden, die nicht irgendeine Steuerfunktion ausüben, sondern die lediglich dazu dienen, mit der Steuergerät-Schnittstelle 10 in Verkehr zu treten. Es sei bemerkt, daß die Tabelle A und die Fig. 9 lediglich ein Beispiel für das Format darstellen, und daß bei einer besonderen Anlage viele andere Privilegzzweige und Kommandosprossen hinzugefügt werden können.

Das programmierbare Steuergerät 9 leitet ein Kommando dadurch ein, daß das Startbit für dieses Kommando auf eins gesetzt wird. Die Startbits sind im Datentabellenabschnitt des Speichers 212 des Steuergerätes gespeichert, und die Steuergerät-Schnittstelle 10 arbeitet derart, um diese Start-Geschehen-Wörter periodisch auszu-lesen und dahingehend zu überprüfen, ob irgendwelche Kommandos auszuführen sind. Es ist dem Anwender überlassen, sein Steuerprogramm so zu schreiben, daß die Startbits zum richtigen Zeitpunkt gesetzt werden, um die gewünschten Funktionen auszuführen.

Wenn die Steuergerät-Schnittstelle 10 feststellt, daß ein Kommandosprossenstartbit gesetzt worden ist, liest sie aus dem Kommunikationssprossenabschnitt 213 des Speichers 212 des Steuergeräts die Kommandodaten für die betreffende Sprosse aus. Die Steuergerät-Schnittstelle 10 bildet dann eine Nachricht und sendet sie an die Bestimmungsstation aus, die in der aktiven Kommandosprosse angegeben ist. Nachdem die Antwort empfangen

worden ist, setzt sie das geeignete "Getan"- oder "Geschehen"-Bit im Start-Geschehen-Wort. Das Steuerprogramm im programmierbaren Steuergerät enthält Befehle, die dieses Ereignis erfassen, und das Steuerprogramm setzt das Startbit für die Kommandosprosse zurück. Die Steuergerät-Schnittstelle setzt dann das "Geschehen"-Bit zurück.

Für die folgende Erläuterung wird insbesondere die Fig. 6 herangezogen. Die periodische Prüfung des Start-Geschehen- oder Start-Getan-Wortes im Speicher 212 des programmierbaren Steuergerätes wird von der Steuergerätabtast- und Nachrichteneinleitungs-Task 123 vorgenommen. Diese Task 123 durchläuft alle 50 ms einmal ihren Zyklus, und während jeder derartigen "Abtastung" überprüft sie ein Start-Getan-Wort und initiiert die angegebenen Nachrichten. Um diese Task beschleunigt auszuführen, sind beträchtliche Mengen an Daten im Speicher 35 der Steuergerät-Schnittstelle gespeichert, und zwar Daten, die den Platz und die Größe der Kommunikationssprossen und der zugeordneten Start-Getan-Wörter betreffen. Diese Daten werden von einer Sprossenprüfroutine RUNGCK erzeugt, die sowohl während des Hochfahrens der Versorgungsenergie der Steuergerät-Schnittstelle 10 als auch immer dann ausgeführt wird, wenn das programmierbare Steuergerät in die Run- oder Durchlauf-Betriebsweise geschaltet wird. Eine Auflistung der Sprossenprüfroutine befindet sich im Anhang B.

Es wird jetzt auch die Fig. 10 zur Erläuterung herangezogen. Die im Steuergerät-Schnittstellen-Speicher gespeicherten Kommunikationssprossendaten werden im folgenden auch mit Kommandosprossenindex bezeichnet. Für jedes in der Steuergerädatentabelle gespeicherte Start-Getan-Wort speichert der Kommandosprossenindex seine 16-Bit-Speicheradressen, wie es bei 220 in der Fig. 10 angegeben ist. Bezüglich der Anzahl der zu versorgenden Start-Getan-Wörter besteht keine Einschränkung, obgleich im allgemeinen ein oder zwei Wörter ausreichend sind. Bis zu acht Kommandosprossen kann man jedem Start-Getan-Wort zuordnen. Für jede dieser Sprossen speichert ein Wort 222 die Anzahl der Zeilen, die die Kommandosprossen im Speicher 212 des Steuergerätes einnimmt, und ein Wort 223 speichert einen 3-Bit-Zeiger und bis zu fünf Statusbits. Die die Sprossengröße betreffenden Wörter 222 werden von der Abtasttask 123 benutzt, um schnell in die im Speicher 212 des Steuergerätes gespeicherten Kommandosprossen zu indizieren und um auf diese Weise die benötigten Daten zu erhalten. Der 3-Bit-Zeiger zeigt dabei an, welches der acht Startbits im Start-Getan-Wort dieser besonderen Kommandosprosse zugeordnet ist. Die Statusbits in den Wörtern 223 enthalten ein Bit, das der Abtasttask 123 anzeigt, daß die Kommandosprosse die letzte ist, welche einem besonderen Start-Getan-Wort zugeordnet ist, ein Bit, das anzeigt, daß die Kommandosprosse in den Kommunikationssprossen 213 die letzte ist, und ein Bit, das der Abtasttask 123 anzeigt, daß das Startbit für die Kommandosprosse bereits zuvor erkannt worden ist und gerade verarbeitet wird.

Die Sprossenprüfroutine extrahiert auch Daten aus den Privilegzzweigen der Kommunikationssprossen 213 und speichert sie im Speicher 35 der Steuergerät-Schnittstelle. Diese Daten enthalten eine Liste der Stationsnummern, die das Privileg haben, in den Speicher 212 des Steuergerätes zu schreiben, zusammen mit Daten, die die Speicheradresse des jeder aufgelisteten Station zugeordneten Privilegzzweiges angeben.

Die Steuergerätabtast- und Nachrichteneinleitungs-Task 123 wird alle 50 ms ausgeführt, um zu bestimmen, ob das programmierbare Steuergerät 9 eine Nachricht an eine andere Station des Kommunikationsnetzwerks auszusenden wünscht. Unter Bezugnahme auf die Fig. 6 und 11 wird erläutert, daß die Task 123 immer dann, wenn sie einen Zyklus durchläuft, einen Satz von Befehlen ausführt, die in einem Prozeßblock 225 angegeben sind, um eines der Start-Getan-Wörter aus dem Speicher 212 des programmierbaren Steuergerätes zu lesen. Sie überprüft dann die acht Startbits, um festzustellen, ob irgendwelche von ihnen gesetzt sind. Falls keine gesetzt sind, wird entsprechend der Darstellung in einem Entscheidungsblock 226 eine Subroutine 227 aufgerufen, um das System zum Scheduler 125 zurückkehren zu lassen, und zwar durch Erzeugen der durch Software eingeleiteten Unterbrechung.

Ist andererseits ein Startbit gesetzt, soll eine Nachricht ausgesendet werden, und der Kommandobefehl wird aus dem Speicher 212 des Steuergerätes gelesen und überprüft, um festzustellen, ob eine Lese-, Schreib- oder Bitsteuernachricht ausgesendet werden soll. Wenn Daten aus einer bezeichneten Station ausgelesen werden soll, wie es durch einen Entscheidungsblock 228 bestimmt wird, wird die Nachricht dadurch gebildet, daß die übrigen Daten in der Kommandosprosse aus dem Speicher 212 des Steuergerätes gelesen werden, daß ein Kopf oder Kopfteil gemäß dem Kommunikationsnetzwerkprotokoll angebracht wird und daß die Nachricht in einem Nachrichtenpufferabschnitt des Speichers 35 der Schnittstelle gespeichert wird. Wenn bei einem Entscheidungsblock 229 ein Schreibkommando festgestellt wird oder wenn bei einem Entscheidungsblock 230 ein Bitsteuerkommando festgestellt wird, erfolgt die Bildung einer geeigneten Nachricht und ihre Speicherung im Nachrichtenpuffer. Wie es in einem Prozeßblock 231 angedeutet ist, wird dann eine Leitkurs-Subroutine ROUTE aufgerufen, die ausgeführt wird, um die Nachricht zu einer Ausgabeschlange im Speicher 35 zu transferieren. Eine Auflistung der Abtast- und Nachrichteneinleitungs-Task 123 befindet sich im Anhang C.

Die Funktion der Abtast- und Nachrichteneinleitungs-Task 123 besteht darin, Lese-, Schreib- und Bitsteuerkommandos zu formen, die an andere Stationen des Übertragungs- oder Kommunikationsnetzwerks auszusenden sind. Das Format des Lesebefehls sieht wie folgt aus:

```
DEST SRCE CMD STS TRNS ADDRESS SIZE
```

wobei:

DEST ist die Stationsnummer, an die die Nachricht auszusenden ist;
 SRCE ist die Stationsnummer, von der die Nachricht ausgehend wird;
 CMD ist ein 8-Bit-Byte, in dem Bit 7:0 = Nachricht, 1 = Statusbit, Bit 6:0 = Kommando, 1 = Antwortbit, Bit 5:0 = Normal, 1 = Priorität, Bit 0 bis 3: Kommandoart;
 5 STS wird in einer Kommandonachricht nicht verwendet;
 TRNS ist ein 8-Bit-Feld, das man auf irgendeinen Wert setzen kann und das von der Bestimmungstation in ihrer Antwort zurückgegeben wird;
 ADDRESS ist die 16-Bit-Adresse im Bestimmungstationsspeicher des Blocks auszulesender Daten;
 SIZE ist die Größe in Bytes des Blocks auszulesender Daten.

Das Format des Schreibkommandos lautet wie folgt:

DEST SRCE CMD STS TRNS ADDRESS DATA

wobei:

DEST, SRCE, CMD und TRNS sind wie beim Lesekommando;

ADDRESS ist die 16-Bit-Adresse im Bestimmungstationsspeicher, in den der Block von Daten einzuschreiben ist;
 20 DATA sind die einzuschreibenden Daten, die eine Länge von 1 bis 245 Bytes haben können.

Das Format des Bitsteuerkommandos ist wie folgt:

DEST SRCE CMD STS TRNS 0 bis 50-Bit-Kommandos

wobei:

Jedes Bitkommando enthält eine 16-Bit-Adresse eines Wortes in dem Speicher des programmierbaren Steuergerätes, eine 16-Bit-Maske zu setzender Bits und eine 16-Bit-Maske zu löschender Bits.

Der Fluß von Kommandonachrichten und anderen Daten zwischen dem Kommunikationsnetzwerk, dem Programmierpult und dem programmierbaren Steuergerät ist in der Fig. 14 schematisch dargestellt. Die verschiedenen Warteschlangen und Puffer, die in dieser Figur gezeigt sind und auf die in der folgenden Erläuterung Bezug genommen wird, sind Teile des Direktzugriffsspeichers 35, die durch eine Speichermanagementroutine zugewiesen werden. Die Speichermanagementroutine ist in Wirklichkeit ein Satz von Routinen innerhalb des Systemexekutiv- oder Systemausführungsprogramms, das gerufen werden kann, um spezifische Speichermanagementfunktionen auszuführen. Diese Routinen umfassen eine Nimm-Routine GET, die einen Speicherplatzpuffer oder eine Warteschlange beschafft und zuweist, eine Freigaberoutine RELEASE, die einen zugewiesenen Speicherplatz zur allgemeinen Verwendung freigibt, eine Koppel-Routine LINK, die Nachrichtendaten zu einer Warteschlange oder zu einem Puffer koppelt, und eine Entkoppel-Routine UNLINK, die eine Nachricht von einer Warteschlange oder einem Puffer entkoppelt.

Für die folgende Erläuterung werden insbesondere die Fig. 3 und 14 herangezogen. Der Datenfluß zwischen dem Programmierpult und dem programmierbaren Steuergerät geht direkt vonstatten. Die Steuergerät-Treiber-Routine 117 betätigt den Kanal B des SIO 41, um Daten vom Speicher des programmierbaren Steuergerätes zu empfangen und sie zum Programmierpult-Handhaber 120 weiterzuleiten. Das Programmierpult 15 enthält einen Mikroprozessor, der über das Kabel 22 Daten anfordert. Der Programmierpult-Handhaber 120 empfängt diese Anforderungen, prüft sie und gibt sie weiter an die PC-Warteschlange. Die Steuergerät-Treiber-Routine 117 sendet die Anforderungen über das Kabel 23 an das programmierbare Steuergerät. In manchen Fällen erwidert das programmierbare Steuergerät mit Daten, die durch die Steuergerät-Treiber-Routine 117 empfangen werden. Die Routine 117 sendet die Daten an den Programmierpult-Handhaber 120, der sie wiederum über das Serien-Kabel 22 an das Programmier-Pult aussendet.

Wie man der Fig. 14 entnehmen kann, empfängt die Steuergerät-Treiber-Routine 117 von einer Reihe von Quellen Daten, und zwar zum Laden in das programmierbare Steuergerät. Zusätzlich zur Programmierpult-Handhabungsroutine 120 umfassen diese die Steuergerätabtast- und Nachrichtenausführungsroutine 123, die NETX-Routine 122 und die Antwort-Routine 121. Daten von diesen Quellen werden in einem PC-Warteschlangenabschnitt (PC = programmierbares Steuergerät) des Speichers gespeichert, und die Steuergerät-Treiber-Routine 117 liest aus dieser Warteschlange aus.

Die NETX-Routine 122 ist nicht nur eine Datenquelle für die PC-Warteschlange, sondern sie empfängt auch Daten von der Steuergerät-Treiber-Routine 117. Aus den Fig. 6, 12 und 14 kann man entnehmen, daß es die Funktion der NETX-Task 122 ist, Kommandonachrichten auszuführen, wenn sie von einer anderen Station empfangen werden, und Antwortnachrichten zu der Ursprungsstation zurückzusenden. Wenn der Scheduler 125 die NETX-Task 122 aktiviert, setzt er zunächst die Ereignisflagge in seinem Taststeuerblock zurück und stellt dann fest, ob eine Kommandonachricht im Kommandowarteschlangenabschnitt des Speichers 35 auf ihn wartet. Wenn keine Kommandoschicht wartet, wie es durch einen Entscheidungsblock 233 festgestellt wird, springt das System zurück zum Scheduler 125. Andernfalls wird die Nachricht entkoppelt, wie es durch einen Prozeßblock 234 angedeutet ist, und es werden Prüfungen vorgenommen, um festzustellen, ob die sendende Station das Privileg hat, aus dem Speicher des programmierbaren Steuergerätes zu lesen oder in diesen Speicher einzu-

schreiben. Falls dies nicht zutrifft, wie es durch einen Privileg-Entscheidungsblock angedeutet ist, wird eine Antwort erzeugt, die anzeigt, daß die sendende Station nicht privilegiert ist. Andernfalls werden Prüfungen am CMD-Feld der Nachricht vorgenommen, um festzustellen, ob es sich um ein Lese-, Schreib- oder Bitsteuerkommando handelt. Falls ein Lesekommando vorliegt, wie es durch einen Entscheidungsblock 235 festgestellt wird, wird den Daten, die aus dem Speicher 212 des programmierbaren Steuergerätes zu lesen sind, ein Speicherplatz (d. h. der Antwortpuffer) zugewiesen. Entsprechend einem Prozeßblock 237 werden dann die Daten aus dem durch das Lesekommando spezifizierten Steuergerätspeicherplatz des Steuergerätspeichers 212 gelesen und zur Bildung einer Antwortnachricht für die anfordernde Station verwendet.

Handelt es sich bei der Kommandonachricht um ein Schreibkommando oder ein Bitsteuerkommando, was von einem Entscheidungsblock 238 festgestellt wird, werden die Daten in der Kommandonachricht in den spezifizierten Platz des Steuergerätspeichers 212 geschrieben. Unbeschadet des Kommandos wird entsprechend einem Prozeßblock 239 eine Antwortnachricht gebildet und zwecks Rückübertragung zur Ursprungsstation an die Ausgabewarteschlange freigegeben. Erfäßt der Entscheidungsblock ein unerlaubtes Kommando, wird vor dem Aussenden ein Fehlercode in die Antwortnachricht eingesetzt.

Das Format der Antwortnachrichten ist wie folgt, und eine Auflistung der NETX-Task befindet sich im Anhang D.

Lesekommandoantwort:

DEST SRCE CMD STS TRANS DATA

wobei:

DEST	ist die Stationsnummer des Urhebers der Kommandonachricht;
SRCE	ist die Stationsnummer der Station, die die Antwort sendet;
STS	ist ein 8-Bit-Statusbyte, bei dem:
	0 = Kommandoausführung bestätigt
	1 = unerlaubter Befehl
	2 = Aufforderung zur manuellen Bedienung
	3 = reserviert
	4 = Stationsspeicher voll
	5 = Schreibschutzverletzung
	6 = reserviert
	7 = unzustellbare Nachricht
TRANS	ist ein 8-Bit-Feld, das mit dem in der Kommandonachricht identisch ist;
DATA	sind 1 bis 253 Bytes aus dem Steuergerätspeicher gelesener Daten.

Schreibkommandoantwort und Bitsteuerkommandoantwort:

DEST SRCE CMD STS TRANS

Dabei gilt für DEST, SRCE, STS und TRANS das gleiche wie bei der oben erläuterten Lesekommandoantwort.

Wenn eine Antwortnachricht von der Station empfangen wird, die die Kommandonachricht erzeugte, wird die Ereignisflagge im Tasksteuerblock für die Antworttask 121 gesetzt, und die Antwortnachricht wird in einer Antwortwarteschlange gespeichert. Wenn die Antworttask 121 durch den Scheduler 125 aktiviert ist, entkoppelt sie die Antwortnachricht von der Antwortwarteschlange und prüft das STS-Feld, um festzustellen, ob die Kommandonachricht richtig empfangen worden ist. Trifft dies zu, wird für die entsprechende Kommandosprosse das Getan-Bit im Speicher 212 des programmierbaren Steuergerätes gesetzt, und das System kehrt zum Scheduler 125 zurück. Falls es sich um die Antwort auf ein Lesekommando handelt, enthält jedoch die Antwort die angeforderten Daten, und die Antworttask 121 ist tätig, um sie in den bezeichneten Platz des Steuergerätspeichers 212 zu schreiben. Eine Auflistung der Antworttask befindet sich im Anhang E.

Aus den Fig. 3 und 6 kann man erkennen, daß von der Abtask 123 oder der NETX-Task 122 in die Ausgabewarteschlange eingebrachte Nachrichten unter der Leitung des Protokolltreibers 116 und des Datenverbindungstreibers 115 durch den Kanal A des SIO 41 zum Kabel 1 übertragen werden. Gleichermaßen werden über das Kabel 1 vom SIO 41 empfangene Nachrichten von den Treibern 115 und 116 verarbeitet und entweder in der Kommandowarteschlange oder der Antwortwarteschlange gespeichert. Aus der folgenden Erläuterung geht hervor, daß der Datenverbindungstreiber 115 zu einem großen Teil ein Satz von Subroutinen ist, die vom Protokolltreiber 116 aufgerufen werden, um spezifischen Funktionen auszuführen. Die Subroutinen werden daher in Verbindung mit dem Protokolltreiber 116 beschrieben.

Der Protokolltreiber 116 führt vier grundsätzliche Funktionen aus. Als erstes gewinnt sie die Herrschaft über das Kommunikationsnetzwerk, wenn sich eine Nachricht in der Ausgabewarteschlange der Station befindet. Zweitens überträgt sie die Nachricht zur anderen Station des Netzwerks und akzeptiert eine Statusnachricht. Drittens ruft sie andere Stationen des Kommunikations- oder Übertragungsnetzwerkes auf, um einer von ihnen die Herrschaft zu übertragen, wenn sich keine weiteren Nachrichten in der Warteschlange befinden. Viertens empfängt der Protokolltreiber 116, wenn die Station nicht die Herrschaft hat bzw. nicht Hauptstation ist, Nachrichten von den anderen Stationen des Netzwerks und bringt sie in die Kommandowarteschlange oder

Antwortwarteschlange und sendet Statusnachrichten an die sendenden Stationen zurück. Eine Auflistung der Protokolltreiberroutine 116 befindet sich im Anhang F. Eine Auflistung der zugeordneten Datenverbindungstreiberoutine 115 ist im Anhang G zusammengestellt.

Die Funktion des Aussendens von Nachrichten an das Übertragungsnetzwerk für den Fall, daß die Station Hauptstation ist, wird von einem mit "TESTQ" bezeichneten Teil des Protokolltreibers wahrgenommen. Dieser Teil der Routine 116 ist als Flußdiagramm in der Fig. 13A dargestellt. Es wird eine Schleife gebildet, in der die Ausgabeschlange wiederholt überprüft wird, wie es durch einen Entscheidungsblock 250 angedeutet ist. Wenn keine weiteren Nachrichten auszusenden sind, zweigt das System zum Aufrufteil der Protokolltreiberroutine 116 ab. Andernfalls wird eine Nachricht von der Ausgabewarteschlange entkoppelt, wie es durch einen Prozeßblock 251 dargestellt ist. Ein "RETRY"-Zähler (Wieder-Versuch-Zähler) wird dann auf fünf gesetzt, wie es in einem Prozeßblock 252 zu sehen ist. Anschließend wird eine Reihe von Subroutinen im Daten Verbindungstreiber 115 aufgerufen, um die Nachricht zu senden und auf eine Statusnachricht von der Bestimmungsstation zu warten. Diese Reihe von Subroutinen ist in einem Prozeßblock 253 angegeben. Wenn das Aufrufen der Subroutinen beendet ist, wird zunächst gemäß einem Entscheidungsblock 254 überprüft, ob die Statusnachricht rechtzeitig und akkurat empfangen worden ist. Trifft dies zu, wird der Inhalt der Statusnachricht entsprechend einem Entscheidungsblock 255 geprüft, um festzustellen, ob die gerade ausgesandte Nachricht von der Bestimmungsstation richtig empfangen worden ist. Trifft dies zu, wird die gerade ausgesandte Nachricht gelöscht, und das System tritt erneut in die Schleife ein, um zu sehen, ob noch weitere Nachrichten ausgesendet werden sollen.

Wenn während der Übertragung der Nachricht oder der Antwort auf die Nachricht ein Übertragungsfehler auftritt, wie es von Entscheidungsblöcken 254, 255 und 257 festgestellt wird, durchläuft das System die Schleife nochmals, um die Nachricht erneut auszusenden. Wenn nach sechs Übertragungsversuchen immer noch Fehler auftreten, zweigt das System zu einem Entscheidungsblock 258 ab und berichtet der Task, die die Nachricht erzeugte, daß sie nicht zugestellt werden kann. Falls die Nachricht deswegen nicht zugestellt werden kann, weil die Bestimmungsstation keinen hinreichenden Speicherraum zur Verfügung hat, um die Nachricht zu empfangen, zweigt das System am Entscheidungsblock 257 ab, und die Nachricht wird in eine Warteschlange zum erneuten Aussenden zu einem späteren Zeitpunkt gegeben.

Nachdem alle Nachrichten in der Ausgabeschlange der Station ausgesendet worden sind, zweigt das System zu dem mit "SOPOL" bezeichneten Aufrufteil des Protokolltreibers 116 ab. Zur weiteren Erläuterung wird auch die Fig. 13C herangezogen. Der Zweck des Aufrufteils SOPOL besteht darin, andere Stationen des Übertragungsnetzwerks aufzurufen, um zugunsten von einer anderen dieser Stationen die Herrschaft aufzugeben. Entsprechend einem Prozeßblock 260 ruft das System zunächst nach einer mit "POLLIN" bezeichneten Subroutine, um eine Globalaufrufkommandonachricht zu bilden. Diese Nachricht wird an alle Stationen des Netzwerks adressiert und enthält Parameter, die definieren, welche Stationen auf diesen Aufruf antworten können. Anfangs sind diese Parameter so eingestellt, daß es jeder Station möglich ist, auf den Aufruf zu antworten. Stationen, die innerhalb dieser Parameter zu liegen kommen, werden als vom Aufrufkommando "umfaßt" bezeichnet. Unter Bezugnahme auf einen Prozeßblock 261 sei bemerkt, daß das Globalaufrufkommando mit dem gleichen Satz von Verbindungstreiber-subroutinen wie eine Kommando- oder Antwortnachricht ausgesendet wird. Wenn irgendeine an das Netzwerk angeschlossene Station in ihrer Ausgabewarteschlange eine Nachricht hat, wird diese Station (oder Stationen) antworten, indem sie für eine kurze Zeitspanne ihren Senderträger einschaltet. Falls, wie es von einem Entscheidungsblock 262 festgestellt wird, keine Antwort erfolgt, wird die Ausgabeschlange der aufrufenden Station erneut überprüft, um nachzusehen, ob sie irgendwelche auszusendenden Nachrichten hat. Trifft dies entsprechend der Feststellung eines Entscheidungsblocks 263 nicht zu, wiederholt die Station den beschriebenen Aufrufvorgang.

Wenn eine Antwort auf das Globalaufrufkommando empfangen wird, arbeitet das System derart, daß es die Herrschaft auf diejenige von den antwortenden Stationen überträgt, die die höchste Priorität hat. Wie es durch einen Prozeßblock 264 angegeben ist, wird im Globalaufrufkommando ein Prioritätsbit gesetzt, und dann erfolgt das Aussenden, wie es in einem Prozeßblock 265 angegeben ist. Wenn eine der Stationen, die auf das anfängliche Globalaufrufkommando geantwortet haben, eine Prioritätsnachricht hat, antwortet diese Station auf das Prioritätsglobalaufrufkommando. Wenn keine Prioritätsnachrichten auf ihre Übertragung in einer Warteschlange einer Station warten, was durch einen Entscheidungsblock 266 festgestellt wird, wird das Prioritätsbit zurückgesetzt, bevor mit dem Aufrufen fortgefahren wird. In jedem Falle wird dann in eine Schleife eingetreten, die die Anzahl der durch das Aufrufkommando adressierten Stationen fortschreitend reduziert, bis nur noch eine einzige Station adressiert wird.

Zum Auffinden der auf das Aufrufkommando antwortenden einzigen Station mit der höchsten Priorität wird ein Binärsuchverfahren angewendet. Wie es in einem Prozeßblock 267 angegeben ist, wird die Anzahl der aufgerufenen Stationen während jedes Durchlaufes durch die Schleife halbiert, so daß nur die untere Hälfte der Stationen adressiert wird, die nicht bei vorangegangenen Aufrufkommandos eliminiert worden sind. Das "halbierte" Aufrufkommando wird entsprechend der Darstellung in einem Prozeßblock 268 ausgesendet, und, wenn eine Antwort auftritt, kehrt das System in der Schleife zum Prozeßblock 267 zurück, wie es in einem Entscheidungsblock 269 angegeben ist, um die Anzahl der aufzurufenden Stationen erneut zu halbieren.

Wird andererseits keine Antwort empfangen, muß sich die Station, die die Herrschaft wünscht, in der anderen Hälfte befinden, und das System zweigt über einen Prozeßblock 271 zurück, um das Aufrufkommando zu ändern und dabei die andere Hälfte zu adressieren. Dieser Prozeß oder Vorgang des Aufrufens und Verminderns der Anzahl der aufgerufenen Stationen wird fortgeführt, bis gegebenenfalls, wie es durch einen Entscheidungsblock 270 angedeutet ist, nur noch eine einzige Station aufgerufen wird, die dann antwortet. Die aufrufende Station zweigt jetzt ab und wartet auf empfangene Nachrichten (WTRFM), und die einzige aufgerufene Station übernimmt die Herrschaft über das Kommunikations- oder Übertragungsnetzwerk, d. h. sie springt nach TESTQ. Es soll bemerkt werden, daß, wenn die Übertragung der Herrschaft zwischen zwei oder mehreren Stationen

ausgetragen wird, das hier verwendete Suchverfahren diejenige Station aussucht, die die niedrigste Stationsnummer oberhalb der Nummer der gegenwärtigen Hauptstation hat. In bezug auf die Stationsnummern wird somit von einem Schema mit umlaufender Priorität Gebrauch gemacht. Auf das Prioritätsaufrufkommando kann natürlich jede Station antworten, die eine Prioritätsnachricht hat, und sie kann die Herrschaft über alle Stationen gewinnen, die keine Prioritätsnachrichten haben. Das Format des Aufrufkommandos ist im Anhang H dargestellt.

Im folgenden wird insbesondere auf die Fig. 13B Bezug genommen. Wenn die Schnittstelle 10 des programmierbaren Steuergerätes nicht die Hauptschnittstelle des Kommunikationsnetzwerks ist, wartet sie darauf, daß entweder eine Nachricht oder ein Aufrufkommando an sie gerichtet wird. Während dieses Zustands befindet sich die Protokolltreiberoutine 160 in einem Warteprozeß WTRFM, bei dem ein Zeitgeber gesetzt wird und die Station auf eine an sie gerichtete Nachricht wartet, wie es in einem Prozeßblock 275 angegeben ist. Wenn vor Ablauf des Zeitgebers keine Nachrichten empfangen werden, was durch einen Entscheidungsblock 276 festgestellt wird, wird der Zeitgeber zurückgesetzt, und die Station hält Ausschau nach irgendwelchen Nachrichten im Netzwerk. Wenn keine Nachrichten empfangen werden, was von einem Entscheidungsblock 277 festgestellt wird, springt das System nach TESTQ und übernimmt die Herrschaft über das Netzwerk. In den einzelnen Stationen sind die Zeitgeber auf verschiedene Werte eingestellt, so daß sie nicht zum selben Zeitpunkt den Versuch unternehmen, das Netzwerk zu beherrschen. Vielmehr ist es so, daß diejenige Station, deren Zeitgeber zuerst abläuft, die Herrschaft übernimmt und eine Nachricht (entweder Daten oder ein Aufrufkommando) aussendet, das die Zeitgeber in den anderen Stationen zurückstellt.

Wenn eine Nachricht von der Station empfangen wird, die durch den Entscheidungsblock 276 oder durch den Entscheidungsblock 277 bestimmt ist, wird ihr Bestimmungsadressenfeld geprüft, um festzustellen, ob es sich um ein Aufrufkommando handelt. Trifft dies zu (d. h. die Bestimmungsadresse ist 255), was durch einen Entscheidungsblock 278 festgestellt wird, zweigt das System zu einer Routine BNOLD ab, die bestimmt, ob eine Antwort richtig ist. Wenn die Nachricht kein Aufrufkommando ist, wird das Bestimmungsfeld der Nachricht überprüft, um festzustellen, ob sie an diese Stationsnummer gerichtet ist. Falls nicht, was ein Entscheidungsblock 279 bestimmt, zweigt das System nach WTRFM ab, um auf eine andere Nachricht zu warten.

Wenn eine Nachricht von dieser Station empfangen wird, werden Überprüfungen vorgenommen, um festzustellen, ob diese Nachricht ohne Übertragungsfehler empfangen worden ist und ob Speicherraum verfügbar ist, um die Daten in der Nachricht zu speichern. Wenn die Nachricht in geeigneter Weise empfangen werden kann, was ein Entscheidungsblock 280 feststellt, wird die Nachricht zu der Kommandowarteschlange oder der Antwortwarteschlange übermittelt, wie es in einem Prozeßblock 281 angegeben ist. Außerdem wird die geeignete Ereignisflagge im Prozeßsteuerblock gesetzt, und die Software-Unterbrechung wird eingeleitet, so daß in den Scheduler 125 eingetreten wird. Falls die Nachricht nicht in geeigneter Weise empfangen worden ist, was der Entscheidungsblock 280 feststellt, wird ein nicht null Fehlercode erzeugt, und eine den Fehlercode enthaltende Statusnachricht wird zurück zur verursachenden Station geschickt. Nach dem Aussenden der Statusnachricht kehrt das System in jedem Falle zurück, um auf eine andere Nachricht zu warten.

Wenn ein Aufrufkommando von der Schnittstelle 10 des programmierbaren Steuergerätes empfangen wird, erfolgt eine Abzweigung zu einer mit BNOLD bezeichneten Routine, wie es bereits oben erwähnt ist. Das Flußdiagramm dieser Routine ist in der Fig. 13D dargestellt. Ein Satz von Befehlen, die in einem Entscheidungsblock 285 angegeben sind, wird ausgeführt, um zu bestimmen, ob irgendwelche Prioritätsnachrichten auf Aussendung durch die Station warten. Trifft dies nicht zu, wird, wie es in einem Entscheidungsblock 286 angegeben ist, die Ausgabewarteschlange überprüft, um festzustellen, ob irgendwelche Nachrichten auf Aussendung oder Übertragung warten. Ist dies nicht der Fall, geht das System in der Schleife nach WTRFM zurück, um auf den Empfang der nächsten Nachricht im Netzwerk zu warten.

Befindet sich eine Nachricht in der Ausgabewarteschlange der Station, wird durch einen Entscheidungsblock 287 das Aufrufkommando überprüft, um festzustellen, ob es sich um einen Prioritätsaufruf handelt. Trifft dies zu und befinden sich keine Nachrichten in der Prioritätsausgabewarteschlange der Station, geht das System in der Schleife zurück, um auf die nächste Nachricht zu warten. Andernfalls wird entsprechend einem Entscheidungsblock 288 das Aufrufkommando überprüft, um zu bestimmen, ob diese Station von der Aufrufkommandoadresse umfaßt ist. Trifft dies zu, werden Subroutinen in der Datenverbindungstreiberoutine 115 gerufen, wie es in einem Prozeßblock 289 angegeben ist, um positiv oder zwangsläufig auf das Aufrufkommando zu antworten. Diese Antwort ist ein in geeigneter Weise zeitgesteuerter Impuls oder Impulszug vom Sender der Station. Schließlich wird das Aufrufkommando erneut überprüft, um festzustellen, ob es nur eine einzige Stationsnummer adressiert. Trifft dies zu, was von einem Entscheidungsblock 290 festgestellt wird, übernimmt das System die Herrschaft über das Kommunikations- oder Übertragungsnetzwerk und springt nach TESTQ. Andernfalls erfolgt eine Rückkehr in der Schleife nach WTRFM, um es zu ermöglichen, daß weitere Aufrufkommandos durch die Hauptstation ausgesendet werden können.

Es ist augenscheinlich, daß es zu jedem beliebigen Zeitpunkt eine einzige Hauptstation für das Kommunikations- oder Übertragungsnetzwerk gibt. Allerdings wird die durch die Hauptstation ausgeübte Herrschaft gemäß einem vorgewählten Prioritätsschema zwischen den Stationen transferiert.

Jede Station des Übertragungsnetzwerks, die eine auszusendende Nachricht hat, wird letztlich Hauptstation. Sollte in einer Station ein Fehler auftreten, während diese Station die Hauptstation des Übertragungsnetzwerks ist, übernimmt eine andere Station die Funktion der Hauptstation nach Ablauf einer vorgewählten Zeitspanne. Somit ist das Netzwerk nicht von der Betriebsweise irgendeiner einzigen Station abhängig. Die Herrschaft wird zu einer anderen Station transferiert, nachdem durch die laufende oder gegenwärtige Hauptstation ein ordnungsgemäßer Aufrufprozeß ausgeführt worden ist. Es gibt daher keinen Streit um die Herrschaft, der zu nicht voraussagbaren Bedingungen führen könnte.

Es gibt nur einen sehr kleinen Unterschied zwischen der Schnittstelle 10 des programmierbaren Steuergeräts

und den Schnittstellenschaltungen, die verwendet werden, um andere Einrichtungen mit dem Übertragungsnetzwerk zu verbinden. Aus den Fig. 1, 3 und 6 erkennt man, daß der Unterschied zwischen den verschiedenen Schnittstellenschaltungen 8, 10, 13 und 14 in den Steuergerät- und -empfängerschaltungen 47 sowie in der Steuergerät-Treiber-Routine 117 liegt. Der vom USART 55 in der Schnittstelle 10 des programmierbaren Steuergeräts vorgesehene Serien-Kanal wird von den anderen nicht benötigt. Folglich entfällt diese Hardware und auch die zugeordnete Software. Fast alle handelsübliche numerischen Steuerungssysteme, Prozeßsteuerungs- und Rechnersteuerungssysteme machen von einem Standard-Serien-Kanal RS 232 Gebrauch. Die übrigen Steuergerät-Schnittstellenschaltungen 8, 13 und 14 sind daher scheinbar identisch. Bezüglich ihrer Funktionsweise im Übertragungsnetzwerk arbeiten die Steuergerät-Schnittstellen 8, 10, 13 und 14 in der oben erläuterten Weise.

Obgleich es sich bei dem hier erläuterten Schnittstellen-Modul um einen körperlich getrennten Modul handelt, könnte die Schnittstelle auch in das Gerät einbezogen oder integriert sein, dem sie zugeordnet ist. Die Erfindung könnte beispielsweise integraler Teil einer auf Mikroprozessortechnik beruhenden industriellen Anschlußstation sein.

Zusammenfassend enthält ein industrielles Kommunikationsnetzwerk Mikroprozessoren enthaltende Schnittstellen-Schaltungen, von denen jede ein Steuergerät, beispielsweise ein programmierbares Steuergerät, mit einer Seriendatenverbindung hoher Geschwindigkeit verbindet. Jede Schnittstellenschaltung ist mit der Datenverbindung und mit dem zugeordneten Steuergerät verbunden. Jede Schnittstellenschaltung ist in der Lage, von der Datenverbindung Nachrichten zu empfangen, die an das zugeordnete Steuergerät gerichtet sind. Darüber hinaus kann jede Schnittstellenschaltung die Herrschaft über das Kommunikationsnetzwerk erlangen, wenn die die Herrschaft gerade ausführende Schnittstellenschaltung ein Aufrufkommando erzeugt, das anzeigt, daß sie zur Abgabe der Herrschaft bereit ist. Dies hat zur Folge, daß das Kommunikationsnetzwerk auch noch dann funktions- und arbeitsfähig ist, wenn ein oder mehrere Steuergeräte oder eine oder mehrere den Steuergeräten zugeordnete Schnittstellenschaltungen ausfallen oder gestört sind.

Anhang A

IDEFLT	EQU	\$			
	PUSHA	SAVE		ALLE REGISTER	5
	+				
	+				
	+				
	+				
					10
	IN	A, (.LOW.S8251)		WENN UNTERBRECHUNG DURCH USART	
				VERURSACHT GEHE	
	AND	RXR51		AUSFÜHREN PROGRAMMIERPULT	
				TREIBER	
	JP	Z,DISPAT			15
	JP	I8251			
DISPTX	EQU	\$			
	OUT	(.LOW.SWIO),A		ANDERNFALLS SETZE ZURÜCK	
				SOFTWARE-UNTERBRECHUNGSVER-	20
				RIEGELUNG	
	LD	(\$-\$),SP		BEWAHRE SP IM LAUFENDEN TASK-	
				STEUERBLOCK	
CFLAGX	EQU	\$-2			
	LD	HL,FLAGS-1		NIMM ZEIGER ZU FLAGGENTABELLEN	25
DISLP	INC	L		ZEIGE ZUR NÄCHSTEN FLAGGE	
	LD	A,(HL)		NIMM FLAGGE	
	INC	L		ZEIGE ZUR MASKE	
	AND	(HL)		PRÜFE OB FERTIG ZUM DURCHLAUF	
	JP	2,DISLP-DISPTX+		WENN NICHT FERTIG DURCHLAUFE	30
		DISPAT		SCHLEIFE BIS JA	
	DEC	L		NIMM FLAGGENADRESSE	
	SET	4,L		SP IST BEI FLAGGE +16	
	LD	(CFLAG),HL		BEWAHRE NEUEN AKTIVEN TASK-	35
				ZEIGER	
	LD	A,(HL)		NIMM NEUES SP	
	INC	L			
	LD	H,(HL)			
	LD	L,A			40
	LD	SP,HL			
	POPA	RELOAD		TASK REGISTER	
	EI				
	RET			;STARTE ERNEUT TASK	
DISSIZ	EQU	S-DISPTX			45
	DSEG				
DISPAT	DEFS	DISSIZ			
CFLAG	EQU	DISPAT+(CFLAGX-DISPTX)			
	ASEG				50

5	RUNGCK	EQU	\$		
		LD	HL, (PRVST)	WENN DA IST EINE REZEPT-ABLAGE	
		LD	A, H		
10		OR	L	LÖSCHE SIE	
		LD	DE, FCOUNT		
		CALL	NZ, RELE		
		LD	HL, 0		
		LD	(SBFST), HL	NULL-START-BIT-ABLAGE-ADRESSE (ALS EINE FLAGGE)	
15		LD	B, 1	WEISE EINEN BLOCK DER ABLAGE ZU	
		CALL	GET		
		JP	Z, NOMEN		
		LD	(PRVST), HL	BEWAHRE ADRESSE VOM START DER ABLAGE	
20		LD	(SBFAD), HL	BEWAHRE ABLAGEZEIGER	
		INC	HL	INITIALISIERE PRIVILEGABLAGE	
		LD	(HL), OFFH		
		LD	HL, OFFFEH	SETZE ZURÜCK ADRESSE VOM STEUER- WORT	
25		LD	(WRDAD), HL		
		LD	HL, 00200H	PRÜFE TESTCODES	
		LD	(PXPCB+AH), HL		
30		CALL	XPCR	LESE AUS 0002	
		LD	A, B	0002 MUSS 55 SEIN	
		CP	055H		
		JP	NZ, BADTST		
		ADD	C	0003 MUSS AA SEIN	
35		INC	A		
		JP	NZ, BADTST		
		LD	(IX+AL), 0	BERECHNE GRÖSSE VON DATENTABELLE (GETAN DURCH PLC-TREIBER)	
		CALL	XPCR	NIMM ADRESSE VOM STEUERPROGRAMM	
40		LD	HL, (PGADR)		
		LD	(PXPCB+AH), HL		
	SRCHG	LD	HL, 0		
		LD	(EWAD), HL	NULL-FEHLER-WORT-ADRESSE	
		LD	(ACWRD), HL	AKTIVIERE WORT-ADRESSE	
45		RST	TYPEI	NIMM EINEN OPCODE	
		DEFB	ENDT	WENN ENDE VON MEM GEHE WARTEN FÜR PROGRAMM	
		JP	Z, PROGWT		
		CP	GETI	WENN NICHT OPCODE FAHRE FORT MIT SUCHE	
50		JR	NZ, SRCHG		
		CALL	SHUFL	FORMIERE NEU STATIONSADRESSE UND VERGLEICHE	
		JR	NZ, SRCHG		
55		CP	(IY+STNADR-Y)	SUCHE WENN KEINE ÜBEREINSTIMMUNG	
		JR	NZ, SRCHG		
		RST	TYPEI	NÄCHSTES SOLLTE SEIN ADRESSE VOM STATUSWORT	
		DEFB	GETT		
60		JR	NZ, SRCHG		
65					

PS 30 43 894

	LD	(EWAD),DE	WENN SO BEWAHRE SIE	
	RST	TYPEI	NÄCHSTES SOLLTE SEIN ADRESSE VOM	
			AKTIVIERTEN BIT	
	DEFB	XICT		
	JR	NZ,SRCHG		5
	LD	(ACWRD),DE	WENN SO BEWAHRE ADRESSE	
	LD	A,C	UND BITNUMMER	
	LD	(ACBIT),A		
	RST	TYPEI	NÄCHSTES SOLLTE SEIN OTLO2707	10
	DEFB	OTLT		
	JR	NZ,SRCHG		
	RST	TYPEI	NÄCHSTES KANN SEIN:	
	DEFB	OTUT	OTU 02707 WENN DA IST NICHTS	
	JP	Z,SBSCIN		15
	CP	XICT	XIC WENN DA SIND KEINE PRIVILEGIEN	
	JR	Z,SCUM		
	CP	BSTT	ODER START VON PRIVILEGIEN WENN	
	JR	NZ,BADRNX	(D) NICHTS VON DEM OBIGEN:	
			FEHLER	20
SPRIV	RST	TYPEI	DIE NÄCHSTEN DREI MÜSSEN NIMMS	
			SEIN	
	DEFB	GETT		
	JR	NZ,BADRNX		
	CALL	SHUFL	SETZE AUF EIN SENSIBLES FORMAT UM	25
	JP	NZ,BADSTN		
	PUSH	AF		
	CALL	GSBQB	NIMM EIN BYTE VON ABLAGE	
	POP	AF		
	LD	(HL),A	BEWAHRE STATIONSADRESSE	30
	RST	TYPEI		
	DEFB	GETT		
	JR	NZ,BADRNX		
	RST	TYPEI		35
	DEFB	GETT		
	JR	NZ,BADRNX		
	RST	TYPEI	NÄCHSTES KANN SEIN:	
	DEFB	BSTT	BST; GEHE FÜR ANDERES PRIVILEG	
	JR	Z,SPRIV		40
	CP	BNDT	ODER BND; SONST UNERLAUBT	
	JR	NZ,BADRNX		
	RST	TYPEI	NÄCHSTES MUSS SEIN AUSGABEBEFEHL	
	DEFB	OUTT		45
	JR	NZ,BADRNX		
	RST	TYPEI	NÄCHSTES MUSS SEIN:	
	DEFB	BSTT	BST; GEHE NIMM MEHR PRIVILEGIEN	
	CALL	GSBQB	NIMM EIN ENDE VON ABLAGE-BYTE	
	LD	(HL),OFFH		50
	CP	OTUT	OTU; ENDE VON SPROSSEN	
	JP	Z,SBSCIN		
	CP	XICT	SONST MUSS SEIN XIC; START VON	
			KOMMANDOSPROSSEN	
BADRNX	JP	NZ,BADRNG		55
SCOM	LD	HL,(SBFAD)		
	LD	(SBFST),HL	BEWAHRE ADRESSE VON START VON	
			KOMMANDO-ABLAG	60
				65

PS 30 43 894

	LD	HL, (PXPB+AH)	BEWAHRE ADRESSE VON START VON KOMMANDOSPROSSEN
	CALL	DEC2HL	
	LD	(CUMST), HL	
5	NXCOM	BIT	0, D
			START-BIT MUSS SEIN IN UNGERADZÄHLIGEM BYTE
	JP	Z, BADRNG	
	RES	0, D	
10	LD	HL, (WRDAD)	WENN VERSCHIEDEN WORTADRESSE
	AND	A	
	SEC	HL, DE	
	JR	Z, SMWRD	
	LD	HL, (SBFAD)	SETZE ENDE VON BYTE-BIT
15	SET	NDBY, (HL)	
	CALL	GSBQB	UND BEWAHRE ADRESSE VON NEUEM BYTE
	LD	(HL), E	
	CALL	GSBQB	
20	LD	(HL), D	
	LD	(WRDAD), DE	BEWAHRE NEUE WORTADRESSE
	CALL	GSBQB	RESERVIERE RAUM FÜR SPROSSENGRÖSSE
	LD	(STRTBT), HL	BEWAHRE ADRESSE ZEITWEISE
	LD	(IY+RUNGSZ-Y), 0	LÖSCHE SPROSSENGRÖSSENZÄHLER
25	CALL	GSBQB	BEWAHRE BIT-NUMMER
	LD	A, C	
	LD	(HL), A	
	RST	TYPEI	
30	DEFB	XICT	NÄCHSTER BEFEHL MUSS SEIN KOMMANDO XIC
	JR	NZ, BADRNX	
	LD	A, C	PRÜFE AUF LESEN, SCHREIBEN ODER BITKOMMANDO
35	AND	B	
	CP	Z	WENN LESEN ODER SCHREIBEN GEHE PRÜFEN DREI NIMMS
	JR	C, RWCOM	
	JR	NZ, BADRN	WENN NICHT BIT IST UNGÜLTIG
40	;BITSPROSSE KANN ENTHALTEN IRGEND EINE KOMBINATION VON XIC, SIO, BST und BND; BIS EINE AUSGABE AUFTRIFFT		
	BITRNG	RST	TYPEI
		DEFB	XICT
	JR	Z, BITRNG	
45	CP	XIOT	
	JR	Z, BITRNG	
	CP	BSTT	
	JR	Z, BITRNG	
50	CP	BNDT	
	JR	Z, BITRNG	
	CP	OUTT	
	JR	NZ, BADRN	
	JR	NXCMD	DANN GEHE NIMM NÄCHSTES KOMMANDO
55	RWCOM	RST	TYPEI
		DEFB	GETT
	JR	NZ, BADRN	NÄCHSTE DREI MÜSSEN SEIN NIMMS
	RST	TYPEI	

	RST	TYPEI	REST VON SPROSSE KANN FAST IRGENDETWAS SEIN	5
	DEFB	OUTT		
	JR	NZ,BADRN		
NXCMD	LD	A,(RUNGSZ)	BEWAHRE SPROSSENGRÖSSE IN ABLAGE	
	INC	A		
	LD	HL,(STRTBT)	ABLAGE-ADRESSE WAR HIER GESTAPELT	10
	LD	(HL),A		
	RST	TYPEI	NÄCHSTES KANN SEIN;	
	DEFB	OTUT	ENDE VON ABLAGEN	
	JR	Z,SNFB		
	CP	XICT	ODER ANDERE KOMMANDOSPROSSE	15
	JP	Z,NXCOM		
BADRN	JP	BADRNG		
SNFB	LD	HL,SBFAD)	SETZE ENDE VON ABLAGE-BIT	
	SET	NSF,(HL)		
	SET	NDBY,(HL)	UND ENDE VON STEUERWORT-BIT	20
	JP	SBSCIN	GEHE STARTEN ABTASTUNG	
BADTS	JP	BADTST		
	; SUBROUTINE ZUM NEHMEN EINES BYTE			
GSBQB	PUSH	AF		
	LD	HL,(SBFAD)	NIMM ABLAGEZEIGER	25
	INC	L	RÜCKE VOR ZUM NÄCHSTEN BYTE	
	LD	A,L	WENN NICHT ENDE VON BLOCK	
	AND	OOFH		
	XOR	OOFH		
	JR	NZ,ZBT	GEHE XAP-BYTE	30
	OR	(HL)	PRÜFE FÜR NULL-ZEIGER	
	CALL	NZ,GTADBA	WENN NICHT RICHTIG NIMM NÄCHSTEN	
	JR	NZ,ZBT	PUFFER UND NULL-BYTE (DIES SOLLTE NICHT PASSIEREN)	35
	PUSH	DE		
	PUSH	HL		
	PUSH	BC		
	LD	DE,FCOUNT	NIMM EINEN BLOCK	
	LD	B,1		40
	CALL	GET		
	JP	Z,NSBMEN		
	POP	BC		
	PKPTR	GET	ZEIGER ZU NEUEM BLOCK	45
+				
+				
+				
+				
	POP	DE	VERBINDE NEUEN BLOCK MIT DEM ALTEN	50
	LD	(DE),A		
	POP	DE		
ZBT	LD	(SBFAD),HL	BEWAHRE NEUE ABLAGE-ADRESSE	
	LD	(HL),O	ZAP NEUES BYTE	
	POP	AF		55
	RET			
SVSIZ	LD	HL,(STRTBT)	NIMM ADRESSE VOM KLUDGE-SPEICHER	
	LD	A,(RUNGSZ)		
	LD	(HL),A		
	RET			60
	END			

PS 30 43 894

Anhang C

Steuergerätabtast- und Nachrichtenausführungstask

5	SBSCIN	SET	POK, (IY+PXBITS-Y)	SETZE PROGRAMM OK-FLAGGE
		LD	(IY+SBINDX-Y), -1	SETZE ZURÜCK SPROSSENZÄHLER
		LD	HL, (COMST)	INITIALISIERE ADRESSE VON START VON KOMMUNIKATION
10		LD	(SBADR), HL	
		LD	HL, (SBFST)	NIMM ADRESSE VON START VON START-BIT-ABLAG
		LD	(SBFAD), HL	
15		LD	A, H	
		OR	L	
		JR	Z, PROGWT	WENN KEINE START-BITS WARTE AUF PROGRAMM
20	SBYTE	EQU	\$	WENN ABTASTZEITGEBER NOCH NICHT AN IST WARTE AUF IHN
		LD	A, (PXTMR)	WENN ZEITGEBER LÄUFT
		INC	A	
		LD	A, 2**FT	WARTE AUF BEENDIGUNG DES INTERVALLS
25		CALL	NZ, WTR1	
		RES	FT, (IY+PFLAG-Y)	SETZE ZEITGEBERBIT ZURÜCK
		TEVENT	PTIME, 1, PX	STARTE ABTASTZEITGEBER (50 MS)
30				
35		CALL	STCK	NIMM PLC-STATUS
		JR	NZ, WSCN	WENN FEHLER PROBIERE SPÄTER
		AND	2**RUN	WENN KEIN DURCHLAUF GEHE WARTE AUF DURCHLAUF
		JR	Z, SCAN	
		CALL	GETFIL	NIMM ABLAGEZEIGER
		LD	E, (HL)	NIMM BYTE-ADRESSE VON ABLAGE
40		CALL	GETFIL	
		LD	D, (HL)	
		EX	DE, HL	
		LD	(WRDAD), HL	BEWAHRE WORTADRESSE
		LD	(PXPCB+AH), HL	
45		CALL	XPCR	LIES START- UND GETAN-BITS
		JP	NZ, PCBAD	
		LD	(STRTBT), BC	BEWAHRE START- UND GETAN-BITS
		RES	WDN, (IY+PXBITS-Y)	SETZE ZURÜCK GETAN-BIT MODIFIZIERE FLAGGE
50	ICODE	EQU	\$	NIMM UND VERARBEITE NÄCHSTEN SPROSSENZEIGER
		BIT	POK, (IY+PXBITS-Y)	WENN PROGRAMM SCHLECHT GEHE PRÜFE SPROSSEN
55		JP	Z, SCAN	
		INC	(IY+SBINDX-Y)	INKREMENTIERE START-BIT-INDEX
		CALL	GETFIL	NIMM GRÖSSE VON SPROSSE
		LD	A, (HL)	
		LD	(RUNGSZ), A	
60		CALL	GETFIL	NIMM STARTBITFLAGGEN

65

			NUMMER (BITS 0)	
	LD	B, A		5
	LD	A, (STRTBT)	NIMM STARTBITS	
	AND	B	MASKIERE BIT VON INTERESSE	
	JR	NZ, STRTON	SPRINGE WENN STARTBIT EIN	
	; STARTBIT AUS			
	RES	RSTS, (HL)	SETZE ZURÜCK SPROSSEN-AKTIV-BIT	10
	LD	A, (DONEBT)	NIMM GETAN-BITS	
	AND	B	MASKIERE DAS EINE WELCHES WIR WÜNSCHEN	
	JR	NZ, RSDONE	WENN START AUS UND GETAN EIN	15
	JR	ENDRNG	GEHE ZURÜCKSETZEN GETAN-BIT	
	; GETAN-BIT EIN, SETZE ES ZURÜCK			
	RSDONE	LD	A, B	
	CPL		NIMM MASKE	
	AND	(IY+DONEBT-Y)	SETZE GETAN-BIT ZURÜCK	20
	LD	(DONEBT), A		
	SET	WDN, (IY+PXBITS-Y)	SETZE GETAN-BIT MODIFIZIERTES BIT	
	JR	ENDRNG		25
	; STARTBIT EIN			
	STRTON	LD	A, (DONEBT)	
	AND	B	NIMM GETAN-BITS	
			NIMM DAS EINE WELCHES WIR WÜNSCHEN	
	JR	Z, EXRUNG	WENN START EIN UND GETAN AUS	30
	ENDRNG	LD	A, (RUNGSZ)	
			GEHE PRÜFEN WEITERE	
			ADDIERE GRÖSSE ZU NEHMEN	
			ADRESSE VON NÄCHSTER SPROSSE	
	LD	HL, (SBADR)		
	ADD	A	VERDOPPLE ANZAHL VON WÖRTER ZU BYTES	35
	LD	B, A		
	LD	A, L	ADDIERE ÜBERTRAG ZU HOHEM BYTE	
	ADC	0		40
	LD	A, L		
	LD	A, B	ADDIERE N BYTES TO NIEDRIGEM BYTE	
	ADD H			
	LD	H, A		
	LD	A, L	ADDIERE ÜBERTRAG ZU HOHEM BYTE	45
	ADC	0		
	LD	L, A		
	LD	(SBADR), HL		
	LD	HL, (SBFAD)	NIMM ADRESSE VON LAUFENDEM ABLAGE-BYTE	50
	BIT	NDBY, (HL)	WENN NÄCHSTES STARTBIT IST IM SELBEN BYTE GEHE PROCE	
	JR	Z, ICODE		
	BIT	WDN, (IY+PXBITS-Y)	WENN GETAN-BIT WAR MODIFIZIERT SCHREIBE ES ZURÜCK	55
	JR	Z, NOWRIT		
	LD	HL, (WRDAD)	NIMM ADRESSE VON GETAN-BITS	60
	LD	(PXPCB-AH), HL		


```

        LD      A, (DONEBT)          SCHREIBE GETAN-BITS
        LD      (IX+DH), A
        CALL    XPCW
5  NOWRIT  JP      NZ, PCBAD
        LD      HL, (SBFAD)          NIMM ADRESSE VON STARTBIT-
                                      BESCHREIBUNG ERNEUT
        BIT     NSF, (HL)           WENN ENDE NICHT ENDE VON ABLAGE
                                      GEHE ZU TUN MEHR BYTES
10         JP      Z, SBYTE
        JP      SBSCIN             SONST INITIALISIERE UND
                                      STARTE ERNEUT ABLAGE
; SUBROUTINE ZU NEHMEN UND INKREMENTIEREN EINEN ZEIGER
        DUCH DIE START-BIT-ABLAG
15  GETFIL  LD      HL, (SBFAD)      NIMM ABLAGEADRESSE IM HL
        CALL    INCBLK             INKREMENTIERE ZUM NÄCHSTEN WORT
        LD      (SBFAD), HL        BEWAHRE BIS ZUM NÄCHSTENMAL
        RET
; ZEITABLAUF-ROUTINEN
20  PTIME  SET     FT, (IY+PFLAG-Y)
        SWI
        RET
; ROUTINE ZUM INTERPRETIEREN EINER SPROSSE UND SENDEN VON
25  KOMMANDONACHRICHTEN WENN START-BIT GESETZT
        EXRUNG  EQU      $
        BIT     RSTS, C            WENN NACHRICHT BEREITS
                                      GESENDET RICHTIG KEHRE ZURÜCK
        JR      NZ, ENDRNG
30         LD      HL, (SBADR)      NIMM ADRESSE VON SPROSSE
        LD      (PXPB+AH), HL
        LD      H, C
                                      BEWAHRE CODE ENTHALTEND
                                      BITNUMMER
        CALL    GTAD               LIES ZUERST BEFEHL
35         JP      NZ, BADPC
        LD      A, H
                                      VERIFIZIERE DASS SPROSSE
                                      STARTET MIT ERWARTETEM XIC
        OR      XIC
40         LD      H, A
        CP      C
        JR      NZ, MEMCHG
        BIT     O, D               PRÜFE AUF UNGERADE ADRESSE
        JR      Z, MEMCHG
        RES     O, D
45         PUSH   HL
        LD      HL, (WRDAD)        PRÜFE AUF DIESELBE ADRESSE
        AND     A
        SBC     HL, DE
50         POP    HL
        JR      NZ, MEMCHG
        CALL    TYPE              LIES UND ÜBERPRÜFE KOMMANDO-
                                      BEFEHL
        CP      XICT
55         JR      NZ, MEMCHG
        RES     PXP, (IY+PXBITS-Y) KOPIERE PRIORITÄTSFLAGGE AUS
                                      OPCODE UM ZU BEWAHREN
        BIT     Z, C
60
65

```

JR	Z, \$+6		
SET	PXP, (IY+PXBITS-Y)		
CALL	SHUFL	ERZUEGE BESTIMMUNGSSTATIONS-	
		ADRESSE UND BEWAHRE	5
JR	NZ, MEMCHG		
LD	(IY+DSTAD-Y), A		
LD	A, C	GEHE ZUM KOMMANDOPROZESSOR	
AND	B		
JR	Z, WRITE		10
DEC	A		
JR	Z, READ		
DEC	A		
JP	Z, BIT		
JP	BADRNG		15
MEMCHG	LD HL, 2**MCH		
	JP SERB		
READ	CALL RWPCOM	NIMM PARAMETER VOM PLC	
	JP NZ, SIZER		20
	; A=GRÖSSE VON DATEN, C=STATIONSADRESSE, DE=SRCE-ADRESSE, (SP) = DEST-ADRESSE		
	CP -RSIZE	WENN DATEN WOLLEN NICHT PASSEN IN RESTLICHEN SPEICHER SETZE FEHLER	25
JP	NC, SIZER		
PUSH	AF	BEWAHRE GRÖSSE VON DATEN	
LD	A, RSIZE+3	NIMM KOMMANDORAHMENGROSSE VOM NACHRIKTEHKOPF UND ORDNE ZU SPEICHERPUFFER FÜR NACHRICHT	30
POP	1		
POP	AF	NIMM DATENGROSSE	
POP	DE	NIMM BESTIMMUNGSADRESSE	
LD	(HL), E	BEWAHRE IN NACHRICHT	35
INC	HL		
LD	(HL), D		
INC	HL		
LD	(HL), A		
JP	SEND	GEHE ZU SENDEN NACHRICHT	40
WRITE	CALL RWPCOM	NIMM PARAMETER	
	JP NZ, SIZER		
PUSH	AF	BEWAHRE GRÖSSE	
ADD	RSIZE+2	ADDIERE ÜBERKOPF	
JP	C, SIZER	WENN WILL NICHT PASSEN SETZE FEHLERFLAGGE	45
LD	(PXPCB+AH), DE	BEWAHRE ADRESSE VON DATEN	
CALL	FORMHD	VOM NACHRIKTEHKOPF UND ORDNE ZU SPEICHERPUFFER FÜR NACHRICHT	50
DEFB	0		
POP	AF	NIMM GRÖSSE UND BESTIMMUNG ZURÜCK	
POP	DE		
LD	(HL), E	BEWAHRE BESTIMMUNGSADRESSE IN NACHRICHT	55
INC	HL		
LD	(HL), D		
INC	HL		
LD	B, A	ERSTELLE GRÖSSE	60

	CALL	PLCRB	BEWEGE DATEN VOM PLC ZUM NACHRICHTENPUFFER
	JP	NZ,BADEPC	
	JP	SEND	
5	;SUBROUTINE ZUM HOLEN VON LESE- ODER SCHREIB-PARAMETERN VON PC-KOMMUNIKATIONSSPROSSEN		
	RWPCOM	PUSH BC	
	CALL	GTAD	NIMM FERNADRESSE VOM PLC
10	CALL	NTRNS	ÜBERSETZENICHT ADRESSE
	POP	HL	STECKE FERNADRESSE HINTER RÜCKKEHRADRESSE
	PUSH	DE	
	PUSH	HL	
15	CALL	GTAD	NIMM ÖRTLICHE ADRESSE
	CALL	NTRNS	
	EX	DE,HL	
	CALL	GTAD	NIMM ENDE-ADRESSE
	CALL	NTRNS	
20	EX	DE,HL	
	LD	A,H	BERECHNE BLOCKGRÖSSE (SUBTRAHIERED VON LH)
	SUB	D	
25	INC	A	
	INC	A	
	LD	H,A	
	LD	A,L	
	SBC	E	
30	PUSH	AF	
	LD	A,D	ÜBERSETZE ADRESSE ZURÜCK NACH PC
	RRCA		
	RL	D	
35	RLA		
	RLA		
	LD	D,A	
	POP	AF	
	LD	A,H	GIB GRÖSSE ZURÜCK IN A
40	POP	BC	
	RET		
	;SUBROUTINE ZUM ÜBERSETZEN EINER INTERNEN PC-ADRESSE IN EINE FÜR EIN LOGISCHES NETZWERK VERSTÄNDLICHE ADRESSE		
45	;EIN: ÜBERSETZTE ADRESSE IN ED ;AUS: NICHTÜBERSETZTE ADRESSE IN ED		
	NTRNS	EQU \$	
	LD	A,D	
50	RRA		
	RRA		
	RR	D	
	RR	D	
55	RLC	D	
	RET		
	;SUBROUTINE ZUM NEHMEN SPEICHERPUFFER FÜR NACHRICHT UND ZU SEINER INITIALISIERUNG		
60			
65			

PS 30 43 894

;IN: A=BLOCKGRÖSSE IN BYTES
;RUF: PARAMETER=KOMMANDOBYTE

FORMHD	LD	B, A	BEWAHRE GRÖSSE UND STATIONS- ADRESSE	5
	PUSH	BC		
	CALL	NBLK	BERECHNE ANZAHL DER BLÖCKE	
	PUSH	DE	BEWAHRE SRCE-ADRESSE	
	LD	DE, 0COUNT	NIMM SPEICHER	
	CALL	GET		
	JP	Z, NOMEM		10
	LD	(CMDBUF), HL	BEWAHRE ADRESSE VOM KOMMANDO- PUFFER	
	POP	DE	NIMM PARAMETER ZURÜCK	
	POP	BC		15
	INC	HL	ADRESSIERE WARTEZÄHLER	
	INC	HL		
	LD	(HL), B	BEWAHRE RAHMENGRÖSSE	
	INC	HL		
	MV	(HL), (IY+DSTAD-Y)	BESTIMMUNGSADRESSE	20
+				
+				
	INC	HL		
	MV	(HL), (STNADR)	SRCE-STATIONSADRESSE	25
+				
+				
	INC	HL		
	EX	(SP), HL	NIMM RÜCKKEHRADRESSE	
	LD	A, (HL)	NIMM KOMMANDOCODE	30
	INC	HL		
	EX	(SP), HL		
	LD	(HL), A	KOPIERE KOMMANDOCODE	
	BIT	PXP, (IY+PXBITS-Y)	KOPIERE PRIORITÄTSBIT	35
	JR	Z, \$+4		
	SET	PRIFLG, (HL)		
	INC	HL		
	LD	(HL), 0	NULLSTATUS	
	INC	HL		
	MV	(HL), (SBINDX)	KOPIERE SPROSSENNUMMER	40
+				
+				
	INC	HL		
	RET			
;BITBEFEHL				45
BIT	LD	A, 40*6	NIMM GENUG RAUM FÜR 40 BITKOMMANDOS	
	CALL	FORMHD		
	DEFB	2		
	LD	(IY+RG-Y), 0	LEITE EIN ANZAHL DER BITKOMMANDOS	50
NXBIT	CALL	TYPE	NIMM BIT-OPCODE	
	JP	NZ, BADPC		
	CP	BSTT	LASSE ABZWEIGBEFEHLE UNBERÜCKSICHTIGT	55
	JR	Z, NXBIT		
	CP	BNDT		

PS 30 43 894

	JR	Z, NXBIT	
	CP	XICT	WENN NICHT XIC
	JR	Z, BTIN	
5	CP	XIOT	ODER XIO
	JP	NZ, ENDR	GEHE BEENDEN SPROSSE
	CALL	NTRNS	ÜBERSETZENICHT ADRESSE
	LD	HL, (CMDBUF)	NIMM ADRESSE VON START VOM DATENFELD
10	SET	2, L	
	MV	(IY+RF-Y), (IY+RG-Y)	NIMM ANZAHL VON BITKOMMAN- DOS IN DATENABLAG
15	XCMD	INC (IY+RF-Y)	SCHLÄNGELE KOMMANDOZÄHLER
		DEC (IY+RF-Y)	
	JR	Z, NWCMD	SPRINGE WENN ADRESSE NICHT BEREITS IM PUFFER
			SONST DEKREMENTIERE ZÄHLER
20	DEC	(IY+RF-Y)	VERSUCHE ALS NÄCHSTES OB WORT- ADRESSE NICHT PASSEND IST
	LD	A, E	
	CP	(HL)	
	JR	NZ, NFAD	
25	CALL	INCBLK	
	LD	A, D	
	SUB	(HL)	
	AND	OFEH	
	JR	NZ, NFAD1	
30	GSPC	EQU \$	SETZE BIT IN KOMMANDO
	LD	A, C	NIMM BEFEHL
	AND	OF8H	WENN XIC SETZE BIT, XIO SETZE BIT ZURÜCK
	CP	XIC	
35	JR	Z, SET	
	CALL	INCBLK	BEWEGE ZEIGER ZUM RÜCKSETZEN DER MASKE
	CALL	INCBLK	
40	SET	BIT 0, D	WENN NIEDRIGES BYTE BEWEGE ZEIGER NOCHMALS
	CALL	Z, INCBLK	
	CALL	Gmask	NIMM BIT-MASKE
	OR	(HL)	SETZE BIT
	LD	(HL), A	
45	JR	NXBIT	GEHE VERARBEITE NÄCHSTES BIT
	CALL	INCBLK	ÜBERSPRINGE BITKOMMANDO
	CALL	INCBLK	
	CALL	INCBLK	
50	CALL	INSBLK	
	CALL	INCBLK	
	CALL	INCBLK	
	JR	XCMD	
	NWCMD	EQU \$	GEHE PROBIERE NÄCHSTES KOMMANDO
55			NIMM RAUM FÜR ANDERES BIT- KOMMANDO
	INC	(IY+RG-Y)	INKREMENTIERE ZAHL DER BIT- KOMMANDOS
	LD	A, (IY+RT-Y)	WENN MEHR ALS 40 ZEIT ZUM STERBEN
60			
65			

CP	41		
JP	NC, SIZER		
CALL	INCBK	NIMM EIN BYTE	
LD	(HL), E	BEWAHRE BITADRESSE	5
CALL	INCBK		
LD	(HL), D		
CALL	INCBK		
PUSH	HL	NIMM UND MACHE NULL	
LD	(HL), 0	4 WEITERE BYTES	
CALL	INCBK	BEWAHRE ADRESSE VON MASKEN	10
LD	(HL), 0		
CALL	INCBK		
LD	(HL), 0		
CALL	INCBK		15
LD	(HL), 0		
POP	HL		
JR	GSPC	NIMM ADRESSE DER ERSTEN MASKE	
; ENDE DER BITSPROSSENVERARBEITUNG			
ENDR	LD A, (IY+RG-Y)		20
RLCA		NIMM KOMMANDOZÄHLER	
ADD	(IY+RG-Y)	MULTIPLIZIERE MIT 6	
RLCA			
LD	HL, (CMDBUF)	NIMM ADRESSE VON PUFFER	25
INC	L		
INC	L		
ADD	Rsize		
LD	(HL), A	BEWAHRE GRÖSSE VON RAHMEN	30
; ENDE VON KOMMANDOSPROSSENVERARBEITUNG			
SEND	LD HL, (SBFAD)	NIMM STARTBITABLAGEZEIGER	
	SET RSTS, (HL)	SETZE NACHRICHTEN-MARSCHROUTE-STATUS-BIT	
ON	B		
+	LD HL, (CMDBUF)	SENDE NACHRICHTENPUFFER ZUR AUSGABEWARTESCHLANGE	35
	LD DE, OUTQ		
	CALL LINK		
	JP ENDRNG		
BADSTN	LD HL, 2**WAE		40
	JR SERB		
BADTST	8QU \$		
; SETZE EIN BIT (HARDWARE-EINGRIFF)			
	LD SP, PSTK		45
	JP PROGWT		
NOMEM	EQU \$		
NSBMEN	LD HL, 2**NME		
	JR SERB		
BADRNG	EQU \$		50
	LD HL, 2**BRE		
	JR SERB		
SIZER	EQU \$		
	LD HL, 2**BSE		55
	JR SERB		
BADFC	EQU \$		
PCBAD	LD HL, PCE	KOMMUNIKATIONSFEHLER MIT PC	
SERB	LD SP, PSTK	LADE ERNEUT STAPELZEIGER	60
CALL	SETTER		
	JP SCAN		
	END		

5	NETX	LD	SP, NSTK	
		LD	HL, NFLAG	SETZE ZURÜCK TASK-STEUER- BLOCK-FLAGGEN
10		RES	FQ, (HL)	
		RES	FT, (HL)	
		LD	HL, NPFLAG	NIMM ADRESSE VON PRIORITÄTS- BITS
15		LD	A, (CMDQ)	SETZE NETX-PRIORITÄTSBIT WENN IRGENDETWAS AUF KOMMANDO- WARTESCHLANGE
20		AND	A	
		RES	NPR, (HL)	
		JR	Z, NONP	
		SET	NPR, (HL)	
25	NONP	LD	DE, CMDQ	NIMM EINE NACHRICHT VON KOMMANDOWARTESCHLANGE UND ENTKOPPELE SIE
		CALL	UNLINK	
		JR	Z, WNETX	WENN LEER GEHE WARTEN FPR NACHRICHT
30		LD	(CMDFRM), HL	SONST BEWAHRE SPEICHERADRESSE VON NACHRICHT
		PUSH	HL	
		POP	IX	
35		LD	HL, ICOUNT	ZEIGE EIGENTÜMER VON NACHRICHT AN
		LD	(OWN), HL	
		LD	HL, JTB	NIMM ADRESSE VON KOMMANDO- SPRING-TABELLE
40		LD	A, (IX+CMD)	NIMM CMD/STS-BYTE VON NACHRICHT
		BIT	CMRP, A	
		JR	NZ, IGM	
45		AND	OOFH	PRÜFE AUF GÜLTIGES KOMMANDO
		CP	(JTB1-JTB)/2	
		JR	NC, ILLCMD	
50		JP	TJMP	SPRINGE ZUM AUSFÜHREN ANGEZEIGTES KOMMANDO
	JTB	EQU	\$	
		DEFW	WRITE	
55		DEFW	READ	
		DEFW	BIT	
	JTB1	EQU	\$	
	LOOP	EQU	\$	
60				
65				

PS 30 43 894

CALL	XFR		TRANSFERIERE NACHRICHT ZUM AUSSSENVERKEHR (WENN NICHT BEREITS)	
CALL	SWITCH		TAUSCHE STATIONSNUMMERN AUS IN	5
SET	CMRP, (IX+CMD)		BESTIMMUNGS- UND SRCE-FELDERN SETZE ANTWORBIT IM NACHRICHTEN- STS-FELD	
ON	B			
OUT	(.LOW.LED30), A			
LD	HL, (CMDFRM)		KOPPELE ANTWORTNACHRICHT ZUR AUSGABE	10
LD	DE, OUTQ			
CALL	LINK		WARTESCHLANGE UND SCHLEIFE ZURÜCK ZUM PROZESS	15
JP	NETX		WEITERE BEFEHLSNACHRICHTEN	
;ERZEUGE VERSCHIEDENE FEHLERCODES				
ILLCMD	LD	A, 1*16	FEHLERFORMAT ODER BYTE-ZÄHLWERT UNERLAUBT	20
	JR	SRP		
NVERF	LD	A, 2*16	PC-SPEICHERTRANSFER NICHT VERIFIZIEREN	
	JR	SRP		
HRDWR	LD	A, 3*16	PC NICHT EINGESTÖPSELT ODER ANDERER MANUELLER EINGRIFF	25
	JR	SRP		
MISC	LD	A, 4*16	WIEDERHOLTE TRENNUNG, LIES FEHLER, ODER LASS ZEIT ABLAUFEN	30
	JR	SRP		
SNDOK	XOR	A		
SRP	LD	IX, (CMDFRM)		
	LD	(IX+STS), A	BEWAHRE STATUS	35
	LD	(IX+SIZ), RSIZE	SIZE = ANTWORTGRÖSSE	
	LD	DE, OWN)	GIB EXTRABLÖCKE ZURÜCK ZUM EIGENTÜMER DES RAHMENS	40
	LD	HL, (CMDFRM)		
	CALL	CHOP		
;LIES KOMMANDO				
READ	EQU	\$		45
	LD	A, (IX+DAT+2)	NIMM ANZAHL ZU LESENDER BYTES	
	ADD	RSIZE	BERECHNE GRÖSSE VON ANTWORT- NACHRICHT WENN GRÖßER ALS 256	
	CP	-DST-1	IST SIE ZU GROSS WENN SIE NICHT PASST IN 17 BLÖCKE IST SIE ZU GROSS	50
	JR	NC, ILLCMD		
	LD	(IX+DAT+3), A	BEWAHRE GRÖSSE VON ANTWORT	
CALL	GETRD		WEISE ZUR BILDUNG DER ANTWORT SPEICHERRAUM ZU	55

PS 30 43 894

	JP	Z, NETX	NACHRICHT, UND WENN NICHT VER- FÜGBAR, GIB LESEBEFEHL IN AUFSCHUBWARTESCHLANGE
	MV	(IX+SIZ), (IX+DAT+3)	SETZE NEUE GRÖSSE
5	CALL	GTPTS	GIB ZEIGER ZUM PCB UND DATEN IN PUFFER
	CALL	GTAD	NIMM STEUERGERÄTSPEICHERADRESSE VON KOMMANDONACHRICHT
10	LD	B, (HL)	NIMM ANZAHL ZU LESENDER BYTES VOM KOMMANDO
	DEC	L	DATEN WERDEN DATENFELD VON KOMMANDO ÜBERLAGERN
15	DEC	L	
	CALL	PLCRB	LIES BLOCK VON DATEN VOM PC
	CALL	NSTCR	ÜBERPRÜFE STATUS
	JR	LOOP	
20	; SCHREIBE KOMMANDO		
	EQU	\$	
	WRITE	EQU	\$
	LD	A, (IX+SIZ)	BERECHNE GRÖSSE VON DATENBLOCK
25	SUB	RSIZE+2	RECHNUNGSTRAGEND FÜR ZWEI ADRESS-BYTES
	LD	B, A	
	CALL	GTPTS	NIMM ZEIGER FÜR BLOCK IO BIS PC
30	CALL	GTAD	NIMM ADRESSE VOM SCHREIBKOMMANDO
	CALL	PLCWB	SCHREIBE BLOCK IN PC
	CALL	NSTCK	PRÜFE STATUS
	JP	SNDOK	ERWIDERE MIT OK-ANTWORTNACHRICHT
35	; BITSTEUERBEFEHL		
	PBIT	EQU	\$
	BIT	EQU	\$
	LD	A, (IX+SIZ)	BERECHNE ANZAHL VON BYTES IM DATENBLOCK
40	SUB	RSIZE	
	LD	L, A	RICHTE EIN ZUM DIVIDIEREN DURCH 6
	LD	H, 0	
45	LD	A, 6	
	CALL	DIV	
	LD	A, H	WENN GRÖSSE NICHT VIELFACHES VON 6 UNGÜLTIG
50	AND	A	
	JP	NZ, ILLCMD	
	LD	B, L	ANDERNFALLS B=ANZAHL VON BIT- KOMMANDOS
55	BNWD	CALL	GTPTS
		CALL	GTAD
		RES	CWR, (IX+CN)
		CALL	PCIOH
		CALL	NSTCK
			ÜBERPRÜFE STATUS

PS 30 43 894

LD	D, (HL)	NIMM SETZMASKE VOM PUFFER	
CALL	INCBLK		
LD	C, (HL)		
CALL	INCBLK		
LD	A, (HL)	NIMM TEIL VON RÜCKSETZMASKE	5
CPL			
LD	E, A		
CALL	INCBLK		
LD	A, (NTPCB+DH)	BERECHNE HOHES BYTE	10
OR	D	SETZE BITS	
AND	E	SETZE BITS ZURÜCK	
LD	(NTPCB+DH), A		
LD	A, (HL)	NIMM LETZTEN TEIL VON RÜCKSETZMASKE	15
CPL			
LD	E, A		
CALL	INCBLK		20
LD	A, (NTPCB+DL)	BERECHNE NIEDRIGES BYTE	
OR	C	SETZE BITS	
AND	E	SETZE BITS ZURÜCK	25
LD	C, A	BEWAHRE	
SET	CWR, (IX+CN)	SCHREIBE ERSTES BYTE	
CALL	NSTCK		
CALL	INCAHL	INKREMENTIERE PC-ADRESSE	30
LD	(IX+DH), C	SCHREIBE ZWEITES BYTE	
CALL	PCIOH		
CALL	NSTCK		
CALL	INCAHL		35
DJNZ	BNWD	WIEDERHOLE BIS GETAN	
JP	SNDOK	DANN SENDE ANTWORTNACHRICHT	
; SUBROUTINE ZUM GEWINNEN VON ZUSÄTZLICHEM SPEICHERRAUM BENÖTIGT ZUM AUSFÜHREN EINER AUSLESUNG IN: A=GEWÜNSCHTE GRÖÖE DES RAHMENS			40
; AUS: WENN NZ: OPERATION ERFOLGREICH			
WENN Z: NICHT HINREICHEND VIEL RAUM VERFÜGBAR, NACHRICHT IN AUFSCHUBWARTESCHLANGE PLAZIERT			45
GETRD	CALL	NBLK	BERECHNE ANZAHL BENÖTIGTER BLÖCKE
DEC	B		BERECHNE ANZAHL ZUSÄTZLICHER BENÖTIGTER BLÖCKE
JR	Z, NOXTR		WENN 1 HABEN WIR BEREITS 1 (KOMMANDOBLOCK)
LD	DE, OCOUNT	NIMM EXTRABLÖCKE	55
CALL	GET		
JR	NZ, GOTM	WENN OK FAHRE FORT	
LD	HL, (CMDFRM)	SONST BRINGE KOMMANDONACHRICHT IN AUFSCHUBWARTESCHLANGE	60

```

LD      DE,DEFRQ
CALL    LINK
XOR     A
5      RET
      RETURNIERE NULL KEIN SPEICHER-
      PLATZ

      GOTM  PKPTR  GET
      +
      +
10     +
      +
      LD      HL,(CMDFRM)
      LD      DE,OOFH
      ADD     HL,DE
15     LD      (HL),A
      KOPPLE NEUE BLÖCKE ZUM KOMMAN-
      DOBLOCKENDE

20     NOXTR  DI
      BEWEGE KOPFBLOCK VOM INNENVER-
      KEHR ZUM AUSSENVERKEHR

      DEC     (IX+OCOUNT-Y)
      EI

25     LD      HL,OCOUNT
      LD      (OWM),HL
      ZEIGE EIGENTÜMER DES RAHMENS AN

      OR      OFFH
      RET
30     RETURNIERE OK-STATUS

; SUBROUTINE ZUM ERSTELLEN VON ADRESSZEIGERN
; WIRKUNG: IX=ADRESSE VON PCB
;          FORTFAHR-ADRESSE IST GESETZT IM PCB
35     ;
;          VERIFIZIER-BIT IST GESETZT IM PCB
;          HL=ADRESSE VOM START VOM DATENFELD IM RAHMEN

GPTPS   EQU    $
LD      IX,NTPCB
40     SET     CVF,(IX+CN)
LD      HL,NETPCT
      NIMM ADRESSE VOM PCB
      SETZE VERIFIZIER-BIT
      NIMM ADRESSE VON FORTFAHR-
      ROUTINE

LD      (INTPCB+CTAD),HL
LD      HL,(CMDFRM)
45     NIMM ADRESSE VON DATEN IM RAHMEN

      SET     3,L
      RET

50     ; SUBROUTINE ZUM LADEN EINER ADRESSE VOM KOMMANDONACHRICHTEN-
      PUFFER IN DEN PCB

      GTAD    EQU    $
      MV      (IX+AH),(HL)
55     +
      +
      CALL    INCBK

60

65

```

PS 30 43 894

LD	A, (HL)	NIMM NIEDRIGES BYTE VON ADRESSE	
RRCA		ÜBERSETZE ADRESSE IN DAS PC-FORMAT	
RL	(HL)		5
RLA			
RLA			
RR	(HL)		
LD	(IX+AL), A	BEWAHRE IM PCB	10
JP	INCBK		
;SUBROUTINE ZUM ÜBERPRÜFEN DES STATUS RETURNIERT VOM PCIOH, PLCRB ODER PLCWB			15
;EIN: A=STATUS			
NSTCK	AND A RET Z	WENN NULL FAHRE FORT	20
POP	HL	ANDERNFALLS WIRF RETURNIER-ADRESSE WEG	
BIT	1, A		25
JP	NU, NVERF	LESEN NICHT VERIFIZIERT	
BIT	5, A		
JP	NZ, HRDWR	PLC NICHT EINGESTÖPSELT	30
JP	MISC	LESE-FEHLER	
;SUBROUTINE ZUM FORTFAHREN NETX			
NETPCT	SET FD, (IY+NFLAG-Y)	SETZE GERÄTEREIGNISFLAGGE	35
	SWI HIT	SCHEDULER	
+ RET			
;NETX-ZEITGEBER-EREIGNIS-SUBROUTINE			40
NWAKE	SET FT, (IY+NFLAG-Y)	SETZE NETX-ZEITGEBER-EREIGNIS-FLAGGE	
RET			
;SUBROUTINE ZUM TRANSFERIEREN EINER NACHRICHT VON DER LAUFENDEN ZUWEISUNG ZUM AUSSENVERKEHR			45
XFR	EQU \$		
	LD IX(CMDFRM)		
PUSH	AF		50
LD	A, (IX+SIZ)	NIMM GRÖSSE VOM RAHMEN	
CALL	NBLK		

PS 30 43 894

	LD	HL, (OWN)	NIMM ADRESSE VOM LAUFENDEN EIGENTÜMER
	DI		
5	ADD	(HL)	GIB BLÖCKE FREI
	LD	(HL), A	
	LD	HL, OCOUNT	NIMM ADRESSE VOM OCOUNT
10	LD	A, (HL)	NIMM BLÖCKE VOM AUSSENVERKEHR
	SUB	B	
	LD	(HL), A	
	EI		
15	LD	(OWN), HL	ZEIGE LAUFENDEN EIGENTÜMER DER NACHRICHT AN
	POP	AF	
20	RET		
	END		
25			
30			
35			
40			
45			
50			
55			
60			
65			

Anhang E

Antwort-Task

WREPLY	LD	A, 2**FQ	WARTEN AUF WARTESCHLANGENEINTRITT	5
	CALL	WTFRT		
REPLY	RES	FQ, (IY+REFLAG-Y)	SETZE EREIGNISFLAGGE ZURÜCK	10
	LD	DE, REPLYQ	WENN NICHTS IN ANTWORTWARTE-	
	CALL	UNLINK	SCHLANGE GEHE WARTEN AUF ETWAS	
	JR	Z, WREPLY		
	LD	(RPFRAD), HL	BEWAHRE ANTWORTNACHRICHTEN-	15
	LD	IX, RPPCB	ADRESSE	
	SET	B, L		20
	DEC	D		
	LD	B, (HL)	NIMM TRANSPARENTES BYTE	
	INC	B	(SPROSSENINDEX)	25
	LD	HL, (SBFST)	NIMM ADRESSE VOM START VON DER	
	LD	A, H	START-BIT-ABLAGEN	
	OR	L		30
	JP	NZ, BADREP	WENN KEINE STARTBITS IGNORIERE	
	LD	DE, (COMST)	NIMM ADRESSE VOM START VON	35
			SPROSSEN	
NBYTE	CALL	INCBLK	INKREMENTIERE ABLAGEZEIGER	40
+	MV	(ADDONE), (HL)	BEWAHRE ADRESSE VOM STEUERWORT	
+				
	CALL	INCBLK		45
	MV	(ADDONE+1), (HL)		
+				
+				
NRUNG	CALL	INCBLK	STOSSE ZEIGER	50
	DEC	B		
	JR	Z, FNDRNG	DURCHLAUFE SCHLEIFE BIS INDEX=0	55
	LD	A, (HL)	ADDIERE GRÖSSE VON ÜBERSPRUNGE-	
	ADD	A	NER SPROSSE	
	JR	NC, \$+3	VERDOPPEL SPROSSENGRÖSSE	
	INC	E	(BYTES=2*WÖRTER)	60
				65

	ADD	D	
	LD	D, A	
	JR	NC, \$+3	
	INC	E	
5	CALL	INCBK	SPRUNG ZUM NÄCHSTEN BYTE IN DER ABLAGE
10	BIT	NSF, (HL)	WENN ENDE VON ABLAGE DER INDEX IST UNGÜLTIG
	JP	Z, BADREP	
	BIT	NDBY, (HL)	WENN ENDE VON BYTE GEHE HOLE EIN ANDERES
15	JR	NZ, NBYTE	
	JR	NRUNG	ANDERNFALLS NIMM NÄCHSTEN SPROSSENDESKRIPTOR
20	FINDRNG	CALL	INCBK
	LD	A, (HL)	NIMM UND BEWAHRE GETAN-BIT- NUMMER
	LD	(DONNM), A	
25	BIT	RSTS, A	TESTE SPROSSENSTATUSBIT
	JP	Z, BADREP	WENN SPROSSE WAR INAKTIV IGNORIERE NACHRICHT
30	EX	DE, HL	BEWAHRE SPROSSENADRESSE
	LD	(RPPCB+AH), HL	
	CALL	GTAD	NIMM START-BIT-OPCODE
	JP	NZ, PCBAD	
35	LD	A, (DONNM)	VERGLEICHE MIT ABLAGE
	AND	07	
	OR	XIC	
	CP	C	
40	JP	NZ, BADRNG	
	LD	HL, (ADDONE)	STELLE SICHER STARTBITADRESSE IST DIE SELBE
45	INC	H	
	AND	A	
	SBC	HL, DE	
	JP	NZ, BADRNG	
50	CALL	TYPE	ÜBERPRÜFE KOMMANDO-CODE
	JP	MZ, PCBAD	
	CP	XICT	
	JP	NZ, BADMEN	
55	LD	HL, (RPFRAD)	NIMM NACHRICHTENADRESSE

LD	A, (HL)	BEWAHRE GRÖSSE VON NACHRICHT	5
SUB	RSIZE		
JP	C, BADRSZ		
LD	B, A		
INC	L		
INC	L		
CALL	SHUFL	NIMM STATIONSADRESSE VON SPROSSE	10
CP	(HL)	VERGLEICHE MIT NACHRICHTEN-QUELLE	
JP	NZ, BADREP	WENN NICHT DIE SELBE AKZEPTIERE NICHT	15
INC	L		
LD	A, C	MASKIERE KOMMANDOCODE VON SPROSSE	
AND	B		
XOR	(HL)	VERGLEICHE MIT KOMMANDO IN NACHRICHT	20
AND	OOFH		
JP	NZ, BADREP		
LD	A, (HL)		25
CP	O41H	WENN NICHT LESEN AKZEPTIERE ANTWORT	
JR	NZ, SETDNE		
INC	L	WENN STATUS NICHT OK LESE NICHT	30
LD	A, (HL)		
AND	A		
JR	NZ, SETDNE		
INC	L	NIMM ADRESSE VON DATEN	35
INC	L		
PUSH	HL		
CALL	RWPCOM	NIMM ADRESSEN UND GRÖSSE VON SPROSSE	40
POP	HL	(BESEITIGE FERNADRESSE RETURNIERT AUF DEN STAPEL)	
JP	NZ, BADRSZ		
CP	B	WENN NICHT DAS SELBE WIE RAHMEN SETZE FEHLER	45
JP	NZ, BADRSZ		
POP	HL	NIMM DATENADRESSE	50
LD	(RPPCB+AH), DE	BEWAHRE BESTIMMUNGSADRESSE	
CALL	PLCWB	SCHREIBE BLOCK	
JR	NZ, PCBAD		
SETDNE	LD HL, (ADDONE)	NIMM ADRESSE VOM GETAN-BIT	55
LD	(RPPCB\$AH), HL		
CALL	PCINC2	INKREMENTIERE ADRESSE VOM FERN-FEHLER-BIT	
CALL	XPCR	LIES ES	60

65

	JR	NZ,PCBAD	
	PUSH	BC	BEWAHRE ES
5	LD	HL,(RPFRAD)	NIMM STATUS
	SET	Z,L	
	INC	L	
	INC	L	
	LD	H,(HL)	
10	LD	C,(IY+DONNM-Y)	NIMM BIT-MASKE
	CALL	Gmask	
	LD	L,A	
15	POP	BC	NIMM FEHLER-BIT-DATEN
	LD	A,H	ÜBERPRÜFE FERNSTATUS
	AND	OFOH	
	JR	ZRG00D	SPRINGE WENN OK
20	LD	A,L	SONST SETZE FEHLER-BIT
	OR	C	
	LD	C,A	
	JR	RBAD	
25			
	RG00D	LD	A,L
		CPL	
		AND	C
30		LD	C,A
	RBAD	LD	A,H
		AND	OOFH
		JR	Z,LG00D
35		LD	A,L
		OR	B
		LD	B,A
		JR	LBAD
40			
	LG00D	LD	A,L
		CPL	
		AND	B
		LD	B,A
45			
	LBAD	PUSH	BC
		LD	(IX+DH),B
		CALL	XPCW
		CALL	PCINC
50			
		POP	BC
		LD	(IX+DH),C
		CALL	XPCW
		JR	NZ,PCBAD
55			
		LD	DE,(ADDONE)
		LD	(RPPCB+AH),DE
		CALL	XPCR
60		JR	NZ,PCBAD
65			

	LD	A,B	SETZE GETAN-BIT	
	OR	L		
	LD	(RPPCB+DH),A	SCHREIBE ES ZURÜCK	5
	CALL	XPCW		
	JR	NZ,PCBAD		
CHUKBF	LD	SP,RSTK	SETZE STAPELZEIGER ZURÜCK	
	LD	HL,(RPFRAD)		
	LD	DE,ICOUNT	GIB PUFFER AUF LEERLAUF FREI	10
	CALL	RELE		
	JP	REPLY	GEHE PROBIERE EIN ANDERES	
PCBAD	OUT	(.LOW.PCLED1),A	SCHALTE EIN PC-FEHLER-LICHT-	15
	TEVENT	PCLOFF,40,PL	EMISSIONSDIODE	
	LD	HL,PCE	SCHALTE SIE IN EINER SEKUNDE AUS	
	JR	SERBIT	KOMMUNIKATIONSFEHLER MIT PC	
BADRNG	LD	HL,BRE	SPROSSE STIMMT NICHT MIT START-	20
	SET	BADR,(IY+PXBITS-Y)	BITABLAG E ÜBEREIN	
	JR	SERBIT	SETZE SCHLECHTES SPROSSENBIT	
BADMEN	EQU	\$		25
BADREP	LD	HL,BME	KEIN START-BIT SPROSSE INAKTIV,	
	JR	SERBIT	FALSCH E ADRESSE	
BADRSZ	LD	HL,BSE	GRÖSSE IN NACHRICHT ODER SPROS-	30
	JR	SERBIT	SE UNGÜLTIG ODER UNPASSEND	
SERBIT	CALL	SETTER		35
	JR	CHUKBF	SCHREIBE FEHLER-CODE IN PC	
	END			40
				45
				50
				55
				60
				65

Anhang F

Protokolltreiberoutine

5	TESTQ	EQU	\$	
	+	ON	5	
		+	OFF	4
10		CALL	UPOLL	PRÜFE PROGRAMMIERPULT
		LD	HL, PTCLF	NIMM ADRESSE VON PROTOKOLL-FLAGGEN
15		LD	A, (OUTQ)	WENN PRIORITÄTSAUSGABESCHLANGE NICHT LEER SPRINGE NACH SPX
		AND	A	
20		JR	NZ, SPX	
		BIT	PX, (HL)	WENN PRIORITÄTSNACHRICHTEN SENDEFLAGGE GESETZT ABER
		RES	PX, (HL)	PRIORITÄTSAUSGABEWARTESCHLANGE
25		JP	NZ, S1PPOL	LEER SPRINGE NACH S1PPOL UM
		OR	(IY+OUTQ-Y+3)	AUFZURUFEN PRIORITÄTSSTATIONEN
				SONST PRÜFE GEWÖHNLICHE WARTE-
30		JP	Z, SOPOL	SCHLANGE
				WENN LEER STARTE GEWÖHNLICHEN AUFRUF
		JR	LVPX	
35	SPX	SET	PX, (HL)	SETZE PRIORITÄTSNACHRICHT
				SENDE FLAGGE
	LVPX	LD	DE, OUTQ	NIMM EINE NACHRICHT VON DER
40		CALL	UNLINK	AUSGABEWARTESCHLANGE
		LD	(TXFRAD), HL	BEWAHRE NACHRICHTENSPEICHER-
				ADRESSE FÜR SPÄTER
45		INC	L	
		INC	L	
		MV	(FRSZ), (HL)	BEWAHRE ORIGINALNACHRICHTEN-
				GRÖSSE
50	+			
	+	INC	L	
		MV	(TXDST), (HL)	BEWAHRE NACHRICHTENBESTIMMUNGS-
				STATIONSNUMMER
55	+			
		LD	(IY+LKRTY-Y), 6	INITIALISIERE RETRY-ZÄHLER
				(RETRY=NOCIMALIGER VERSUCH)
	RETDX	DEC	(IY+LKRTY-Y)	DEKREMENTIERE RETRY-ZÄHLER
60		JR	Z, CHUKBF	WENN GETAN RETURNIERE ZUM
				SENDER
65				

PS 30 43 894

CALL	CDWAIT	WARTE BIS TRÄGER AUS	
CALL	TXON	BEFÄHIGE SENDER	
LD	HL, (TXFRAD)	NIMM RAHMENADRESSE	
CALL	TXFRM	SENDE RAHMEN	5
CALL	TXOF	SCHALTE SENDER AUS	
LD	HL, (TXFRAD)	NIMM ADRESSE VON NACHRICHT	
INC	L		
INC	L		
MV	(HL), (FRSZ)	SPEICHERE ORIGINALNACHRICHTEN-GRÖSSE ERNEUT	10
LD	(IY+SCTBUF-Y+15), 0	STELLE SICHER SCTBUF IST RICHTIG ABGESCHLOSSEN	15
TEVENT	STSRTO, 3, LK	GESTATTE IM MITTEL 63 MS FÜR STATUSANTWORT VON BESTIMMUNGSTATION	
LD	HL, SCTBUF	NIMM SPEICHERADRESSE VON NOTIZ-BLOCK-PUFFER FÜR STATUSANTWORT	25
CALL	RXFRM	GEHE WARTEN AUF ANTWORTNACHRICHT	
XRSTS	JR	Z, TSTS	
LD	HL, (TXFRAD)	FALLS KEINE HARDWARE-ÜBERTRAGUNGSFEHLER GEHE TESTEN SOFTWARE-ST	30
INC	L	SONST FORDERE BESTIMMUNG AUF DIE ANTWORTNACHRICHT ERNEUT AUSZUSENDEN	35
INC	L		
LD	(HL), 2		
JR	RETXD		
TSTS	LD	HL, SCTBUF+DST	40
LD	A, (HL)	NIMM ZEIGER IN DEN PUFFER	
INC	A	WENN AUFRUFKOMMANDO IRGENDJEMAND IST AUFGEWACHT UND VERSUCHT DAS STEUER AN SICH ZU REISSEN, WIR LEHNEN UNS DAHER LIEBER ZURÜCK WERDEN ABER DIE ERSTEN SEIN DIE AUFWACHEN	45
JP	Z, WTRFM1		
INC	L		
LD	A, (TXDST)	WENN IN DER AUSGESENDETEN NACHRICHT DIE QUELLENSTATIONSNUMMER NICHT MIT DER BESTIMMUNGSTATIONSNUMMER ÜBEREINSTIMMT, NEHMEN WIR AN JEMAND IST AUFGEWACHT	50
XOR	(HL)		
JP	NZ, WTRFM1		55

PS 30 43 894

	INC	L	SONST
	LD	A, (HL)	NIMM STATUS-BYTE
5	BIT	CMST, A	WENN NICHT STATUS-ANTWORTNACH- RICHT GEHE WARTEN
	JP	Z, WIFRM	
10	BIT	STFUL, A	WENN PUFFER IN BESTIMMUNGS- STATION VOLL GIB NACHRICHT IN WARTEN-WARTESCHLANGE
	JR	NZ, WAITBF	
	AND	O3FH	WENN REST VOM STATUS NICHT
15	JR	NZ, RETXD	NULL VERSUCHE ERNEUT (RETRY) AUSZUSENDEN NACHRICHT
	JR	RELOQ	SONST GIB FREI PUFFERRAUM ZUGE- WIESEN DER AUSGESENDETEN NACHRICHT
20	WAITBF	LD HL, (TXFRAD)	DEKREMENTIERE RETRY-ZÄHLER
	INC	L	
	DEC	(HL)	
25	LD	B, 1	
	;JR Z, RETBF	WENN GETAN RETURNIERE ZUM SENDER	
	INC	L	SONST
	LD	DE, WAITQ	TUE PUFFER AUF WARTE-WARTE- SCHLANGE
30	LD	HL, (TXFRAD)	
	CALL	LINK1	
	JP	TESTQ	SPRINGE UND SIEHE NACH OB IRGENDEINE WEITERE NACHRICHT AUF'S AUSSENDEN WARTET
35			
	CHUKBF	LD B, 2	GRUND=NICHT ZUSTELLBARE NACH- RICHT
40	RETBf	LD HL, (TXFRAD)	RETURNIERE NACHRICHT ZUM SENDER (ÖRTLICH)
45	INC	L	
	INC	L	
	LD	A, (RRSZ)	SPEICHERE ERNEUT ORIGINAL- NACHRICHTENGRÖSSE
50	LD	C, A	
	LD	(HL), A	
	INC	L	
	LD	E, (HL)	TAUSCHE STATIONSNUMMERN IN DEN BESTIMMUNGS- UND SRCE-FELDERN DER NACHRICHT AUS
55			
60			
65			

INC	L			
LD	A, (HL)			
DEC	L			
LD	(HL), A			5
INC	L			
LD	(HL), E			
INC	L			
SET	CMRP, (HL)	SETZE ANTWORT-BIT IM KOMMANDO-BYTE		10
INC	L			
LD	(HL), B	BEWAHRE STATUS		15
LD	A, C	NIMM GRÖSSE DER RETURNIERTEN NACHRICHT		
CALL	NBLK			
LD	HL, OCOUNT	BEWEGE VOM AUSSENVERKEHR ZUM INNENVERKEHR		20
DI	(HL)	GIB SPEICHERBLÖCKE ZUM AUSSEN-VERKEHR		25
LD	(HL), A			
DEC	HL			
LD	A, (HL)	NIMM SIE VOM INNENVERKEHR		
SUB	B			30
LD	(HL), A			
EI				
LD	HL, (TXFRAD)			
LD	DE, REPLYQ			35
CALL	LINK	SENDE NACHRICHT ZUR ANTWORT-WARTESCHLANGE		
SET	FQ, (IY+RFLAG-Y)	EREIGNISANTWORTTASK		40
SWI				
+	JP	TESTQ	BRINGE UND SIEHE NACH OB WEITERE NACHRICHTEN AUFS AUSSENDEN WARTEN	45
; ROUTINE ZUR FREIGABE VON AUSGABENACHRICHTEN ZUR LEERLAUF-WARTESCHLANGE				
RELOQ	LD	HL, (TXFRAD)	NIMM ADRESSE VON PUFFER	
	LD	DE, OCOUNT	NIMM ADRESSE VOM ZUWEISUNGS-ZÄHLER	50
	CALL	RELE	GIB PUFFER FREI ZUR LEERLAUF-WARTESCHLANGE	
	JP	TESTQ	PROBIERE NÄCHSTE NACHRICHT	55
STSRT0	EQU	\$	STATUS-WARTE-ZEITABLAUF-ROUTINE	
	DI			
	MARK	C		60
				65

```

+
+
LD      A, (LKTMR)      WENN UNTERBROCHEN IM FENSTER
                        IGNORIERE ZEITABLAUF
5      AND      A
      RET      NZ

      OUTS     A, 1, 0    SPERRE SIO-UNTERBRECHUNGEN

10     +
      +
      +

15     +
      +
      +

      EI

      INC      A          NIMM ZEITABLAUF-STATUS
20     JP      XRSTS      GEHE ZUR STATUSANALYSE

;MACHE EINEN PRIORITÄTSAUFRUF IMMER WENN DIE PRIORITÄTSAUSGABE-
;WARTESCHLANGE LEER WIRD

25     S1PPOL  ON      5
      +
      ON      4
      +

30     CALL    POLLIN    INITIALISIERE SCTBUF ZUM
                        AUFRUFEN
      SET      PP, (IY+PRCLF-Y) SETZE PRIORITÄTSAUFRUF-BIT
      CALL    POLL
      JR      NZ, BSRCH   WENN PRIORITÄTSSSTATION ANT-
                        WORTET GEHE ÜBER ZUM BINÄRSUCH-
35     (
                        AUFRUF
      JP      TESTQ      SONST FAHRE FORT MIT DER AUSGABE
                        VON NACHRICHTEN VON DER AUSGABE-
                        WARTESCHLANGE

40     ;RAHMEN-WARTE-ZEITABLAUF-HANDHABUNG

      WTRFM    TEVENT  WTRF1, 10, PT  WENN KEIN KONTAKT IN 1/4
                        SEKUNDEN WERDE NERVÖS
45     RES      NHNT, (IY+PTCLF-Y)  ANTWORTE NUR AUF DEN EIGENEN
                        NAMEN
      JR      WTRFMX

      WTRFM1   TEVENT  WTRF2, 10, PT  WENN KEINE NACHRICHT WÄHREND
50     WTRFM1   TEVENT  WTRF2, 10, PT  DIESES INTERVALLS EMPFANGEN
                        WIRD

      +
      +
      +

55     +
      SET      NHNT, (IY+PTCLF-Y) LIES ALLE NACHRICHTEN
      JR      WTRFMX

60

65

```

			WAHREND EINES ERSTEN 2-S-ZEIT-INTERVALLS	
	SETUP	CSIOA,RDMAIL		5
+				
+				
	DEFB	0		
	DEFB	3		
	DEFB	RXENS+RCRCEN+LN8SR		
RDMAIL	EQU	\$		10
	EI			
	TEVENT	WTFR2,2PT	WARTE FÜR EIN WEITERES 2-S-ZEITINTERVALL	15
+				
+				
+				
+				
	RET			20
WTFR2	LD	HL,WTFR3	WARTE FÜR EIN WEITERES ZEIT-	
	LD	(PTTLC),HL	INTERVALL DAS GLEICH DER	
	MV	(PTTMR),(STNADR)	NUMMER DIESER STATION GESETZT	25
			IST	
+				
+				
	RET			30
WTFR3	DI		LETZTER ZEITABLAUF	
	MARK	C		
+				
+				
	LD	A,(PTTMR)	WENN UNTERBROCHEN IGNORIERE	35
	AND	A		
	RET	NZ		
	OUTS	A,1,0	SPERRE SIO-UNTERBRECHUNGEN	40
+				
+				
+				
+				
+				
+				45
	EI			
	JP	TESTQ	SPRINGE NACH TESTQ ZUR ÜBER-	
			NAHME DER STEUERUNG DES	
			NETZWERKS	50
;				
WTFRMX	CALL	UPOLL		
	CALL	GETRXP	NIMM EMPFANGSPUFFER	
	LD	(RXFRAD),HL	BEWAHRE ADRESSE VOM PUFFER	55

60

65

PS 30 43 894

	BIT	STFL, (IY+PTCLF-Y)	PRÜFE STATUS SENDE FLAGGE
	RES	STFL, (IY+PTCLF-Y)	
	JR	Z, STSNT	SPRINGE WENN KEIN STATUS-BIT ZUR URHEBERSTATION ZU SENDEN IST
5			
	LD	HL, SCTBUF+SI2	NIMM ADRESSE VOM NOTIZBLOCK PUFFER
10	LD	(HL), SSIZE	GRÖSSE DES STATUSRAHMENS
	INC	L	
	MV	(HL), LSTSRC)	BEWAHRE STATIONSNUMMER VON DEM AN DEN ZU SENDEN IST
15	+		
	+		
	INC	L	
	MV	(HL), (STNADR)	UND UNSERE RETURNIERSTATIONS-NUMMER
20	+		
	+		
	INC	L	
25			
	MV	(HL), (LSTSTS)	UND DEN LAUFENDEN STATUS ZWECKS BILDUNG EINER ANTWORTNACHRICHT
	+		
	+		
30	CALL	CDWAIT	
	CALL	TXON	SCHALTE SENDER EIN
	LD	HL, SCTBUF	NIMM ADRESSE VON STATUSANTWORT-NACHRICHT
	CALL	TXFRM	SENDE SIE
	CALL	TXOF	SPERRE SENDER
35			
	STSNT	LD	HL, (RXFRAD)
		OFF	5
	+		
		OFF	4
40	+		
	CALL	RXFRM	WARTE AUF EINE NACHRICHT
	LD	HL, (FRAD)	
45			
	LD	C, A	BEWAHRE STATUS-BYTE
	AND	B4+B5+B6	WENN IRGENDETWAS PASSIERT
			IGNORIERE NACHRICHT
	JP	NZ, WTFRM	
50			
	INC	L	
	INC	L	
	LD	B, (HL)	NIMM NACHRICHTGRÖSSE
	INC	L	
55	LD	A, OFFH	WENN NACHRICHTENBESTIMMUNGSFELD
	CP	(HL)	IST FF DIESE NACHRICHT IST EIN
	JP	Z, BNPOLD	GLOBALAUFRUFKOMMANDO
60			
65			

LD	A, (STNADR)	WENN NICHT EIGENE STATIONS- NUMMER IGNORIERE NACHRICHT	
CP	(HL)		5
JP	NZ, WTRM		
BIT	1, C	WENN PUFFER ÜBERLÄUFT KANN KEINE ERNEUTE AUSSENDUNG ERFOLGEN	10
JR	NZ, NOTQST		
LD	A, B	SONST WENN NACHRICHTENGRÖSSE =2 LIEGT WIEDERAUSZUSENDE NACHRICHT VOR	15
CP	2		
JR	Z, QST		
NOTQST	OFF	5	20
+	ON	4	
+			
INC	L		25
MV	(LSTSRC), (HL)	SONST WIR HABEN EINE NEUE NACHRICHT; BEWAHRE RETURNIER- STATIONSNUMMER	
+			
+			
BIT	BUFL, (IY+PTCLF-Y)	WENN KEIN PUFFER RETURNIERE STATUSNACHRICHT ANZEIGEND EINEN VOLLEN PUFFER	30
BIT	1, C		35
JR	NZ, BADSTS	WENN PUFFER ÜBERLÄUFT RETURNIE- RE STATUSNACHRICHT ANZEIGEND EINEN FEHLER	
LD	A, B	WENN NACHRICHTENGRÖSSE 4 DAS IST ZU SCHLIMM	40
CP	RSize		
JR	C, BADSTS		
LD	HL, (RXFRAD)	NIMM ADRESSE VOM PUFFER FÜR NACHRICHT	45
LD	DE, ICount		
CALL	CHOP	SCHAFTE EXTRASPEICHER FORT	
CALL	ROUTE		50
CALL	LINK	SENDE NACHRICHT ZUR KOMMANDO- WARTESCHLANGE	
LD	H, B	NIMM FLAGGENADRESSE	55
LD	L, C		
SET	FQ, (HL)	SETZE NETX-EREIGNISFLAGGE	
SWI		ERZEUGE SOFTWARE-UNTERBRECHUNG SO DASS SYSTEM WIRD RETURNIEREN ZUM SCHEDULER	60
			65

	RES	BUFL, (IY+PTCLF=Y)	
	XOR	A	NIMM GUTEN STATUS
SVSTS	OR	2**CMST	SETZE STATUSNACHRICHTEN-BIT
5	LD(LSTSTS), A	BEWAHRE STATUS
	SETSTS	SET	
	JP	STFL, (IY+PTCLF-Y)	SETZE GÜLTIGE STATUSFLAGGE
		WTFRM	WARTE AUF NÄCHSTE NACHRICHT
10	BADSTS	LD	
		A, 2**STBAD	
	JR	SVSTS	
	FULSTS	LD	
		A, 2**STFUL	
	JR	SVSTS	
15	QST	LD	
		A, (LSTSRC)	WENN DIE FRAGE NICHT KAM
	CP	(HL)	VOM SENDER DER LETZTEN NACHRICHT
	MV	(LSTSRC), (HL)	
	+		
20	+		
	JR	Z, SETSTS	SONST RICHTIG SETZE STATUS-
			FLAGGE
	LD	A, STWHO	RETURNIERE "WIR KENNEN DICH
25			NICHT"-STATUS
	JR	SVSTS	
	GETRXB	LD	
		HL, (RXFRAD)	NIMM ALTE RXBUF-ADRESSE
30	BIT	BUFL, (IY+PTCLF-Y)	WENN PUFFERFLAGGE GESETZT
	SET	BUFL, (IY+PTCLF-Y)	LASSE SIE GESETZT
	RET	NZ	
	LD	DE, ICOUNT	SONST NIMM NEUEN PUFFER
	LD	B, 17	MIT GRÖSSE 17
35	CALL	GET	NIMM PUFFER
	RET	NZ	RETURNIERE WENN RAUM ERHALTEN
	RES	BUFL, (IY+PTCLF-Y)	SETZE PUFFERFLAGGE ZURÜCK
	LD	HL, SCTBUF	UND VERWENDE NOTIZBLOCK-PUFFER
40	SOPOL	ON	
	+		
		ON	
	+		
45		OFF	
	+		
	CALL	POLLIN	BILDE GLOBALAUFRUFKOMMANDO-
			NACHRICHT
	RES	PP, (IY+PTCLF-Y)	SETZE PRIORITÄTSBIT AUF AUS
50			
	OPOL	CALL	
		POLL	SENDE GLOBALAUFRUFKOMMANDO-
			NACHRICHT
	JR	NZ, CKPPOL	WENN ANTWORT PRÜFE SIE AUF
			PRIORITÄT
55			
	LD	A, (OUTQ)	WENN AUSGABEWARTESCHLANGE NICHT
			LEER GEHE SIE BEDIENEN
	OR	(IY+OUTQ-Y+3)	
60	JP	NZ, TESTQ	

65

PS 30 43 894

JR	OPOL	SONST FAHRE FORT MIT DEM AUFRUFEN BIS ETWAS PASSIERT	
;PRÜFE PRIORITÄT EINES ANTWORTENDEN AUF EINEN ERSTEN AUFRUF			5
;STATIONEN MIT SOWOHL HOHER ALS AUCH NIEDRIGER PRIORITÄT			
;ANTWORTEN AUF EINEN AUFRUF NIEDRIGER PRIORITÄT AUF EINEN			
;AUFRUF HOHER PRIORITÄT ANTWORTET NUR EINE STATION MIT HOHER			
;PRIORITÄT			
CKPPOL	SET	PP, (IY+PTCLF-Y)	SETZE BIT HOHER PRIORITÄT
CALL	POLL		
JR	NZ, BSRCH		WENN PRIORITÄT HOCH GEHE AUSFÜHREN BINÄRSUCHAUFRUF
RES	PP, (IY+PTCLF-Y)		SONST SETZE ZUERST NIEDRIGE PRIORITÄT
;FÜHRE EINEN BINÄRSUCHAUFRUF AUS ZUM BESTIMMEN WELCHE STATION DIE HERRSCHAFT WÜNSCHT			20
;WENN M .NICHT.=1			
;UND ANTWORT=1 DANN M=(M+1)/2			
;UND ANTWORT=0 DANN FS=FS+(M+1)/2; M=M/2			
;WENN M=1			25
; UND ANTWORT=1 DANN GEHE NACH WTRFM			
; UND ANTWORT=0 DANN FS"STNADR+1; M=OFFH			
BSRCH	LD	HL, SCTBUF+IM	NIMM ADRESSE DER INTERNEN GRÖSSE
	LD	A, (HL)	RUFE EXTERNE GRÖSSE=(INTERN+1)/2
	DEC	L	
	ADD	1	
	RRA		
	LD	(HL), A	
	CALL	POLL	SENDE AUFRUFKOMMANDONACHRICHT AUS
	JR	Z, NORSP	SPRINGE WENN KEINE ANTWORT AUF AUFRUFKOMMANDO
RESPON	LD	HL, SCTBUF+XM	NIMM ADRESSE VON M
	DEC	(HL)	WENN M=1 EINE EINZIGE STATION
	JR	Z, WTRFM	HAT GEANTWORTET AUF DEN AUFRUF
	INC	(HL)	TRANSFERIERE HERRSCHAFT ZU DIESER STATION UND WARTE AUF NACHRICHTEN
	INC	L	
	LD	A, (HL)	SONST IM=(IM+1)/2
	ADD	1	
	RRA		
	LD	(HL), A	
	JR	BSRCH	FAHRE MIT BINÄRSUCHAUFRUF FORT
;NIEMAND HAT GEANTWORTET			

PS 30 43 894

NORGP	LD	HL, SCTBUF+FS	BERECHNE FS=FS+XM $(=(IM+1)/2)$
	LD	A, (HL)	
	ADD	(IY+SCTBUF-Y+XM)	
	LD	(HL), A	
5			
	INC	L	NIMM ADRESSE VON IM
	INC	L	
	LD	A, (HL)	NIMM M
10	CP	1	WENN=1 STARTE AUFRUF ERNEUT
	JP	Z, TESTQ	
	SRL	(HL)	SONST BERECHNE M/2
	JR	BSRCH	FAHRE MIT BINÄRSUCHAUFRUF FORT
15	; SUBROUTINE ZUM BILDEN DER GLOBALAUFRUFKOMMANDONACHRICHT		
	POLLIN	EQU	\$
		LD	HL, SCTBUF+SIZ
		LD	(HL), PSIZE
		INC	L
20			GRÖSSE DES AUFRUFES
	LD	(HL), OFFH	SETZE GLOBALADRESSE (D.H. ALLE STATIONEN)
	INC	L	
25	INC	L	
	LD	A, (STNADR)	FS=FS+1
	INC	A	
	LD	(HL), A	
30			
	INC	L	
	LD	(HL), OFFH	NIMM GRÖSSE VON ADRESSENPLATZ
	INC	L	
35	LD	(HL), OFFH	SETZE INTERNE GRÖSSE VARIABLE
	RET		
40	; SENDE DIE AUFRUFKOMMANDONACHRICHT LAUFEND IN SCTBUF		
	POLL	CALL	UPOLL
			ÜBERPRÜFE DAS PROGRAMMIERPULT
	POP	HL	
45	LD	(PRAD), HL	
	LD	HL, SCTBUF	NIMM ADRESSE DER AUFRUFKOMMANDO NACHRICHT
	LD	A, (PTCLF)	NIMM PROTOKOLLFLAGGE
50	AND	1	TRENNE PRIORITÄTSAUFRUFBIT
	LD	(SCTBUF+PRIFLG), A	BEWAHRE IM AUFRUFKOMMANDO
	CALL	CDWAIT	
	CALL	TXON	SCHALTE SENDER EIN
55	LD	HL, SCTBUF	
	CALL	TXFRM	SENDE AUFRUFKOMMANDONACHRICHT
	CALL	TXOF	SCHALTE SENDER AUS
60			
65			

```

; AUFRUFANTWORTROUTINE
; B=NACHRICHTENGRÖSSE
; C=STATUS
; HL=BESTIMMUNGSSTATIONSNUMMER
5
BNPOLD LD      A,B
        CP     PSIZE
        JP     NZ,WTFRM
        INC    L
        LD     A,(OUTQ)
        AND    A
        JR     NZ,MBRSP
        OR     (IX+OUTQ-Y+3)
        JP     Z,WTFRM
        BIT    PP,(HL)
        JP     NZ,WTFRM
        WENN GRÖSSE NICHT RICHTIG
        IGNORIERE NACHRICHT
10
        LD     A,(OUTQ)
        AND    A
        JR     NZ,MBRSP
        OR     (IX+OUTQ-Y+3)
        JP     Z,WTFRM
        PRÜFE NICHTPRIORITÄT Q
        WENN NICHTS AUSZUSENDEN WARTE
        AUF NÄCHSTE NACHRICHT
15
        BIT    PP,(HL)
        JP     NZ,WTFRM
        PRÜFE PRIORITÄTSBIT IN DER
        AUFRUFKOMMANDONACHRICHT
        WENN GESETZT GEHE AUF NÄCHSTEN
        AUFRUF WARTEN
20
        WENN NICHT IN AUFGERUFENER
        GRUPPE ANTWORTE NICHT
30
        INC    L
        SUB    (HL)
        INC    L
        SUB    (HL)
        JP     NC,WTFRM
        WARTEN AUF NÄCHSTE NACHRICHT
35
        CALL   UNRESP
        SCHALTE SENDER EIN FÜR KURZE
        PERIODE
40
        LD     A,1
        CALL   HSLLOT
        WARTEN FÜR EINE HALBE AUFRUF-
        PERIODE
        CALL   TEST
        JR     NZ,RESPN
        PRÜFE AUF EINE ANTWORT
        WENN RETURNIERTE ANTWORT NICHT
        NULL
45
        LD     A,1
        CALL   HSLLOT
        WARTEN FÜR EINE ZWEITE HÄLFTE
        DER AUFRUFPERIODE
50
        XOR    A
        SONST RETURNIERE NULL
55
PRET    JI     (PRAD)
+
+
        RESPN LD     A,1
        CALL   HSLLOT
        OR     OFFH
        JR     PRET
60
        END
65

```

Anhang G

Datenverbindungstreiberoutine

```

5      HSL0T  EQU  $
        LD    HL, SCTBUF+IM+3
        LD    (HL), A
10     INC    L
        LD    A, (STNADR)
        LD    (HL), A
        DEC   L
        DEC   L
15     DEC   L

      TXFRM  LD    BC, OOCFOH      INITIALISIERE ABWECHSELND
                                   REGISTERSATZ AUF SCHNELL
        LD    D, OOFH
        LD    (FRAD), HL        BEWAHRE ADRESSE VOM RAHLEN
        INC   L
        INC   L
        LD    E, (HL)           NIMM GRÖSSE VOM RAHMEN
        INC   L
25     EXX

        POP   HL
        LD    (SACONT), HL      BEWAHRE FORTFAHR-ADRESSE
30     CALL   OPOLL
        DI

        SETUP CSIOA, TXSET
35     +
        +
        DEFB  1+RSTATI
        DEFB  TXINEN+EXINEN
        DEFB  TCRCRS
40     TXSET  EQU  $
        LD    (IY+LMODE-Y), 1
        EXX
45     OUTI
        EXX
                                   KICKE DEN SIO
        SETUP CSIOA, URRS
50     +
        +
        DEFB  UREMRS
        DEFB  RSTATI
55     URRS   EQU  $
        RET
60
65

```

PS 30 43 894

RXFRM	LD	BC, OOCFOH	ERSTELLE ABWECHSELND REGISTER-SATZ	
	LD	DE, OOFOOH		
	LD	(FRAD), HL	BEWAHRE ADRESSE VOM RAHMEN-PUFFER	5
	INC	L		
	INC	L		
	INC	L		
	EXX			10
	POP	HL		
	LD	(SACONT), HL	BEWAHRE FORTFAHR-ADRESSE	
	BIT	NHNT, (IY+PTCLF-Y)	WENN KEIN NACHLAUF FLAGGE IST	15
			1 SETZE NACHLAUFBETRIEBSART-BIT ZURÜCK	
	JR	Z, HUNTA		
	RES	NHNT. (IY+PTCLF-Y)		20
	DI			
	SETUP	CSIOA, RXSET		
+				
+				
	DEFB	0		25
	DEFB	3+ERRRS		
	DEFB	RXENS+SHUNT+RCRCEN+LN8SR		
	DEFB	1+RCRCRS		
	DEFB	RXAPEN		
RXSET	EQU	\$		30
	JR	FLSH		
HUNTA	DI			35
	SETUP	CSIOA, SETHNT		
+				
+				
	DEFB	0		40
	DEFB	3+ERRRS		
	DEFB	RXENS+RCRCEN+ADSRCH+LN8SR+SHUNT		
	DEFB	1+RCRCRS		
	DEFB	RXAPEN		
SETHNT	EQU	\$		45
FLSH	IN	A, (.LOW.DSIOA)	DURCHSPÜLE EMPFÄNGER	
	IN	A, (.LOW.DSIOA)		
	IN	A, (.LOW.DSIOA)		
	IN	A, (.LOW.DSIOA)		50
	LD	A, RCRCRS+ERRRS		
	OUT	(.LOW.CSIOA), A		
	RET			55
;SENDE PUFFER-LEER-UNTERBRECHUNG				
SATXIN	EXX		WECHSELE REGISTER	
	EX	AF, AF1		


```

DEC      E          GETAN WENN ZÄHLWERT ERREICHT
JR      Z,XNDFRM    NULL
5      OUTI        OUTPUT CHAR
JR      NZ,XIRET    WENN ENDE VOM BLOCK BILDE
                        KETTE
10     LD      B,D    NIMM NEUEN BLOCKGRÖSSENZÄHL-
                        WERT (15)
      LD      L,(HL)  NIMM ZEIGER VOM ENDE VOM
                        PUFFER
15     LD      A,L    NIMM ZEIGER H
      OR      C      (DSIOA MUSS SEIN FO; RAM ORG=
                        F000)
20     LD      H,A
      LD      A,L    NIMM ZEIGER L
      AND     C
      LD      L,A
25     XIRET      EXX      RETURNIERE
      EX      AF,AF1
      EI
      RETI
30     XNDFRM    XOR      A
      OUT     (.LOW.CSIOA),A
      LD      A,TXINRS
      OUT     (.LOW.CSIOA),A  SPERRE SENDERUNTERBRECHUNG
35     JR      XIRET
;SPEZIELLE EMPFANGSBEDINGUNGSUNTERBRECHUNG
;TRITT AUF FÜR:
; RX ÜBERLAUFEN      R1 BIT 5
40    ; CRC FEHLER    R1 BIT 6
; ENDE DES RAHMENS R1 BIT 7
SAREIN  PUSHA
45    +
      +
      +
      +
      +
50    LD      (LKSPSV),SP
      LD      SP,LKSTK
      MARK    LK
      +
      +
55    IN      A,(.LOW.DSIOA)  SCHAFFE LETZTES BYTE VOM CRC WEG
      SETUP   CSIOA,REDISI   SPERRE UNTERBRECHUNGEN UND
                        WÄHLE R1
60
65

```

```

+
+
      DEFB 0
      DEFB 1
      DEFB 0
      DEFB 1
REDISI EQU $

      LD      (OX+LKTMR-Y),OFFH HALTE ZEITABLAUF AN
                                     10
      IN      A, (.LOW.CSIOA)      NIMM STATUS
      XOR     B7                    WENN NICHT ENDE VON RAHMEN
      AND     B7+B6+B5              ODER FEHLER
      JR      NZ,RXDNI              RETURNIERE EINFACH
                                     15
      LD      (IY+PTTMR-Y),OFFH SETZE ZURÜCK ZEITABLAUF FÜR
                                     GÜLTIGEN RAHMEN

      EXX
      LD      A,L                    SONST PRÜFE PUFFERADRESSE
      AND     C                      MASKIERE MIT FO
                                     20
      XOR     H                      XOR MIT HOHEM BYTE
      XOR     C                      KOMPLEMENTIERE HOHE ZIFFER
                                     (SCHAFTE F WEG)
      JR      Z,RXOVER              WENN NULL LIEF DER PUFFER IN
                                     DEN PUFFER ÜBER
                                     25

      LD      HL,(FRAD)              NIMM ADRESSE VON GRÖSSE
      INC     L
      INC     L
      DEC     E                      IGNORIERE CRC BYTE
      LD      (HL),E                BEWAHRE GRÖSSE IM RAHMEN
                                     30
      RXDNIZ  XOR     A              RETURNIERE NULLSTATUS
                                     35
      RXDNI   EI
      PUSH    AF
      CALL    UPOLL
                                     40

      POP     AF
      LD      IX,(FRAD)              NIMM ADRESSE VOM RAHMEN
      JI      (SACONT)
                                     45
+
+
      RXOVER  LD      A,B1
      JR      RXDNI
                                     50

;EMPFGANGENE CHAR-UNTERBRECHUNG
SARXIN EXX
      EX      AF,AF1
                                     WECHSELE REGISTER
                                     55
      INI     DO I/O
                                     60
                                     65

```

	JR	NZ, RIRET	WENN NICHT ENDE VOM BLOCK GEHE RETURNIEREN
	LD	B, D	SONST NIMM NEUEN BLOCKZÄHL- WERT
5	LD	L, (HL)	NIMM ZEIGER ZUM NÄCHSTEN BLOCK
	LD	A, L	
	OR	C	NIMM ZEIGER H
10	LD	H, A	
	LD	A, L	NIMM ZEIGER L
	AND	C	
	LD	L, A	
15	RIRET	INC E	ZÄHLE RAHMENGRÖSSE
		EXX	
		EX AF, AF1	
		EI	
20		RETI	

; EXTERNE- UND STATUS-UNTERBRECHUNG
 ; ENTHÄLT DIESE BEDINGUNGEN IM REGISTER 0
 ; BIT 7 ABORT EMPFANGEN
 ; BIT 6 SENDER UNTERLAUFEN
 ; BIT 4 SYNCH-ERFASSUNG
 ; BIT 3 TRÄGER-ERFASSUNG

30	SASTIN	PUSH HL	STOSSE EINFACH EINIGE BIS EINE WAHRE UNTERBRECHUNG VORLIEGT
		PUSH AF	
	XOR	A	NIMM UND SETZE STATUS ZURÜCK
35	OUT	(.LOW.CSIOA), A	
	IN	A, (-LOW.CSIOA)	
	LD	H, A	
	LD	A, RSTATI	
	OUT	(.LOW.CSIOA), A	
40	LD	A, H	
	LD	L, (IY+LMODE-Y)	VERZWEIGE BEI VERBINDUNGS- BETRIEBSART (XMIT=1, RCVE=2, SYNC=3)
45	DEC	L	
	JR	Z, TMO	
	DEC	L	
	JR	Z, SMO	
50	DEC	L	
	JR	NZ, STOUT	WENN KEINE GÜLTIGE BETRIEBSART- ERWIDERUNG
55	CMO	XOR OFFH	GEMACHT CDECT L AUS SIEHT AUS WIE SYNC H AUS
	SMO	AND 2**4	SYNC BETRIEBSART: NUR NACHLAUF AUS UNTERBRECHUNGSKENNUNG
60			
65			

PS 30 43 894

	JR	NZ,STOUT		
	OR	2**4		
	JR	STIN		
TMO	AND	2**6	SENDEBETRIEBSART: UNTERLAUFE	5
	JR	Z,S: OUT	NUR	
;SENDE UNTERLAUF-ERFASSUNGS-ROUTINE AUS				
;TRITT AUF WENN KEINE DATEN DEM SENDER ZUGEFÜHRT WORDEN SIND				
UND ER KEINE DATEN MEHR ZUM AUSSENDEN HAT. ER SENDET AUTOMA-				
TISCH CRC (D.H. WAS IST LOS)				
	EXX		PRÜFE BYTE-ZÄHLER	15
	DEC	E		
	INC	E		
	EXX			
	JR	NZ, TXFAIL	WENN UNTERLAUFEN BEENDE	20
			NACHRICHT JETZT	
	XOR	A	SONST ÄNDERE VEKTOREN	
	OUT	(.LOW.CSIOB),A		25
	LD	A,2		
	OUT	(.LOW.CSIOB),A		
	LD	A,(.LOW.SSIOV)		
	OUT	(.LOW.CSIOB),A		30
	JR	STOUT		
TXFAIL	LD	H,A	ABORT-NACHRICHT UND SCHALTE	
			UNTERBRECHUNGEN AB	35
	LD	A, SNDABT+1		
	OUT	(.LOW.CSIOA),A		
	XOR	A		
	OUT	(.LOW.CSIOA),A		40
	LD	A,H		
STIN	PUSH	BC	STOSSE DIE ÜBRIGEN REGISTER	
	PUSH	DE		45
	PUSH	IX		
	LD	(LKSPSV),SP		
	LD	SP,LKSTK		
	MARK	LK		50
+				
+				
	LD	(IY+LKIMR-Y),OFFH	SETZE ZEITABLAUF ZURÜCK	
	BIT	4,A	PRÜFE AUF SYNC EIN	55
	JP	NZ,SYNCI		
	AND	A		
	JP	RXDNI	RETURNIERE NICHT NULL WENN	60
			UNTERLAUFEN	

PS 30 43 894

```

STOUT  POP  AF
        POP  HL
        EI
        RETI
5
;SPEZIELLE SENDERUNTERBRECHUNG VERWENDET NUR NACH CRC
SATXIX  PUSH
+
+
10
+
+
+
LD      (LKSPSV),SP
LD      SP,LKSTK
15
MARK    LK
+
+
        SETUP
20
+
+
        DEFB  0
        DEFB  1
        DEFB  0
25
TXADN   EQU   $
        SETUP  CSIOB,VCSET      SPEICHERE ERNEUT NORMALEN
                                   VEKTOR
+
+
        DEFB  0
        DEFB  2
        DEFB  ,LOW.NSIOV
35
VCSET   EQU   $
        JP     RXDNIZ
LKRET   LD     SP,(LKSPSV)
40
        POPA
+
+
+
+
45
        EI
        RETI
50
;WARTE AUF SYNC-ERFASSUNGS-ROUTINE
;WARTE AUF TRÄGER-ERFASSUNGS-AUS-ROUTINE
SYNC    TEVENT SYNTMO,2,LK
+
+
55
+
+
        POP    DE
        LD     (SACONT),DE
60
+
65
```

PS 30 43 894

```

RET
;SYNC ZEITABLAUF
SYNMO EQU $
DI
MARK C
+
+
LD A, (LKTMR)
AND A
RET NZ
WENN BEREITS UNTERBROCHEN
IGNORIERE ZEITABLAUF
;SYNC UNTERBRECHUNG
SYNCI SETUP CSIOA, SYNOF
+
+
DEFB 0
DEFB 1
DEFB 0
SYNOF EQU $
JP RXDN1
;SUBROUTINE ZUM WARTEN AUF TRÄGERERFASSUNG AUS
CDWAIT EQU $
POP HL
LD (SACONT), HL
LD (IY+LMODE-Y), 3
DI
SETUP CSIOA, CDE
+
+
DEFB 0
DEFB 1
DEFB EXINEN
DEFB RSTATI
CDE EQU $
CALL TEST
JR Z, SYNCI
RET
TXON DI
SETUP CSIOA, TON
+
+

```

PS 30 43 894

```

5      DEFB 0
      DEFB 5
      DEFB LN8ST+TXENS+SDLCFC+TCRCEN+RTSS
      DEFB 3
      DEFB LN8SR+RXENS
TON    EQU $

      EI
      RET
10
;SUBROUTINE ZUM SPERREN DES SENDERS
TXOF   POP    HL          BEWAHRE RÜCKKEHR- ODER
                        RETURNIERADRESSE
15      LD     (RXCNT),HL

      DI
      LD     (IY+LMODE-Y),Z
20      SETUP CSIOA,SNCS   GIB FLAGGENSUCHE FREI
      +
      +
      DEFB ERRRS
      DEFB 3
25      DEFB LN8SR+SHUNT+RXENS
      DEFB RSTATI
      DEFB 1
      DEFB EXINEN
SNCS   EQU $
30      CALL  SYNCW        WARTE AUF EIN FLAGGEN-CHAR
      SETUP  CSIOA,TOF     SCHALTE SENDER AUS
35      +
      +
      DEFB 0
      DEFB RSTATI+5
      DEFB LN8ST+DTRS+SDLCFC+TCRCEN
40      TOF    EQU $

      CALL  UPOLL
      JI     (RXCNT)       KEHRE ZURÜCK
45      +
      +
UNRESP DI
      SETUP CSIOA,UNR      SCHALTE SENDER EIN
50      +
      +
      DEFB 0
      DEFB 5
      DEFB LN8ST+TXENS+DTRS+SDLCFC+TCRCEN
55      DEFB 3
      DEFB LN8SR+RXENS
UNR    EQU $
60
65

```

	EI				
	RET				
TEST	DI			SUBROUTINE ZUM PRÜFEN TRÄGER-ERFASSUNG FÜR AUFRUF	5
	SETUP	CSIOA, TDCD		BRINGE STATUS-VERRIEGELUNG AUF NEUESTEN STAND	10
+					
+					
	DEFB	0			
	DEFB	RSTATI			
TDCD	EQU	\$			15
	IN	A, (.LOW.CSIOA)		PRÜFE TRÄGER-STATUS	
	XOR	OFFH			
	AND	DCDS			
	EI				
	RET				20
LKINIT	SETUP	CSIOA, LKI1			
+					
+					
	DEFB	0			25
	DEFB	4			
	DEFB	SDLC			
	DEFB	5			
	DEFB	LN8ST+SDLCFC			30
	DEFB	3			
	DEFB	LN8SR			
	DEFB	7+RSTATI			
	DEFB	07EH			
	DEFB	6			35
LKI1	EQU	\$			
	LD	A, (STNADR)			
	OUT	(.LOW.CSIOA), A			40
	SETUP	CSIOB, LK12			
+					
+					
	DEFB	0			45
	DEFB	2			
	DEFB	.LOW.NSIOV			
	DEFB	1			
	DEFB	STSVCT			
LK12	EQU	\$			50
	RET				55
	END				

1.1 KOPFSPROSSE

XXX YYY ZZZZZ 02707
 --- [G] --- [G] --- [] --- (L) ---

GET	XXX
GET	YYY
XIC	ZZZZZ
OTL	02707

XXX IST DIE STATIONSADRESSE DER SCHNITTSTELLE DIE
DIESEN KOMMUNIKATIONSSPROSSENBEREICH ERARBEITEN
WIRD
YYY IST DIE ADRESSE EINES ORTES WO DIE STATION XXX
FEHLERCODES SPEICHERN KANN
ZZZZZ IST DIE ADRESSE EINES BIT DAS DAS AUSSENDEN UND
EMPFANGEN VON NACHRICHTEN DURCH DIE STATION
XXX FREIGIBT ODER SPERT

DIESER ABSCHNITT ENTHÄLT IRGEND EINE ANZAHL VON SPROSSEN. JEDE SPROSSE KANN EINEN ODER MEHRERE ABZWEIGE AUFWEISEN. JEDER ABZWEIG DEFINIERT EINE STATIONSNUMMER UND EINEN SPEICHERBEREICH, ZU DEM DIE BETREFFENDE STATION ZUGRIFF ERLANGEN KANN. JEDER ABZWEIG MUSS MIT EINEM BST BEGINNEN UND ENTHÄLT DREI NIMM-BEFEHLE MIT FOLGENDER BEDEUTUNG:

$$+ \text{---} \overset{\text{AAA}}{[G]} \text{---} \overset{\text{BBB}}{[G]} \text{---} \overset{\text{CCC}}{[G]} \text{---} +$$

```

BST
GET      AAA
GET      BBB
GET      CCC

```

```
AAA   IST DIE ADRESSE EINER STATION
BBB   IST DIE ADRESSE DES ERSTEN WORTES IN EINEM
      BEREICH, ZU DEM DIE STATION AAA ZUGRIFF
      ERLANGEN KANN
CCC   IST DIE ADRESSE DES LETZTEN WORTES IN DIESEM
      BEREICH.
```

BND- UND LEERAUSGABEBEFEHLE KÖNNEN VERWENDET WERDEN, UM DIE SPROSSE FÜR EINEN ANZEIGEBEREICH PASSEND ZU MACHEN. EIN BEISPIEL KANN WIE FOLGT AUSSEHEN:

```

!   AAA   BBB   CCC   GGG   HHH   III   LEER
!--- G --- G --- G ---+--- G --- G --- G ---+---( )---
!
!   DDD   EEE   FFF   !   JJJ   KKK   LLL   !
!--- G --- G --- G ---+--- G --- G --- G ---+

```

```

BST
GET   AAA
GET   BBB
GET   CCC
BST
GET   DDD
GET   EEE
GET   FFF
BND
BST
GET   GGG
GET   HHH
GET   III
BST
GET   JJJ
GET   KKK
GET   LLL
BND
OTE   LEER

```

1.3 KOMMANDOSPROSSEN

DIESER ABSCHNITT DES PLC-2-PROGRAMMS ENTHÄLT IRGEND EINE ANZAHL VON KOMMANDOSPROSSEN. JEDE KOMMANDOSPROSSE STEUERT EINE NACHRICHT, DIE MAN ZU IRGEND EINER DER STATIONEN SENDEN KANN. JEDE KOMMANDOSPROSSE BEGINNT MIT ZWEI XIC-BEFEHLEN UND ENDET MIT EINER LEERAUSGABE. DER KÖRPER DER KOMMANDOSPROSSE BESTEHT AUS DREI NIMM-BEFEHLEN FÜR LESE- UND SCHREIBKOMMANDOS UND IRGEND EINE ANZAHL VON XIC-, XIO-, BST- UND BND-BEFEHLEN FÜR BIT-KOMMANDOS.

1.3.1 LESE- UND SCHREIBKOMMANDOS

```

GGG1W HHHPI JJJ KKK LLL LEER
---[ ]---[ ]---[G]---[G]---[G]------( )---

```

```

XIC   GGG1W
XIC   HHHPI
GET   JJJ
GET   KKK
GET   LLL
OTE   LEER

```

GGG IST DIE ADRESSE DES ERSTEN WORTES EINES 2-WORT-PAARES. DIESES ELEMENT BEINHÄLTET FOLGENDES.

ADRESSENZUORDNUNGEN:

5	<u>FUNKTION</u>	<u>WORT</u>	<u>BIT</u>
	STARTBIT	GGG	1W
	GETAN-BIT	GGG	0W
	FERNFEHLERBIT	GGG+1	1W
10	ORTSFEHLERBIT	GGG+1	0W

HHH IST DIE BESTIMMUNGSSTATIONSNUMMER
P IST DIE NACHRICHTENPRIORITÄT
I ZEIGT DIE AUSZUFÜHRENDE FUNKTION AN:

15 0 SCHREIBE AN BESTIMMUNG
 1 LIES VON BESTIMMUNG

JJJ IST DIE SPEICHERADRESSE IM BESTIMMUNGSSTEUERGERÄT, WO DIE DATEN STARTEN.
20 KKK IST DIE ADRESSE DES URHEBENDEN STEUERGERÄTS, WO DIE DATEN STARTEN.

LLL IST DIE ADRESSE DES URHEBENDEN STEUERGERÄTS, WO DIE DATEN ENDEN!

1.3.2 BITKOMMANDO

30 GGG1W HHHP2 ! ----- LEER
 ---[] ---[] --- ! IRGEND EINE ZULÄSSIGE KOMBINATION! --- () ---
 ! VON XIC, XIO, BST, BND !

35 XIC GGG1W
 XIC HHHP2

- BEISPIEL -

40 BST
 XIC AAAAA
 XIO BBBBB
 BND

45 -USW-

OTE LEER

50 GGG, W UND HHH WURDEN BEREITS ZUVOR DEFINIERT. JEDES XIC IM KÖRPER DER BITSTEUERSPROSSE DEFINIERT EIN IN DIE DATENTABELLE DES BESTIMMUNGSSTEUERGERÄTES EINZUSETZENDES BIT. JEDES XIO DEFINIERT EIN IM BESTIMMUNGSSTEUERGERÄT RÜCKZUSETZENDES BIT.

55

60

65

1.4 BEGRENZUNGSSPROSSE

DIESE SPROSSE MARKIERT DAS ENDE EINES KOMMUNIKATIONSSPROSSEN-
BEREICHES.

02707

(U)-----

OTU 02707

2.0 NETZWERKPROTOKOLFORMATE

2.1 KOMMANDOS

LIES DST SRC CMD STS TNS ADRESSENGRÖSSE
SCHREIBE DST SRC CMD STS TNS ADRESSENDATEN
BIT DST SRC CMD STS TNS ADRESSENMASKEN

FEHLERERKENNUNG:

SCHLEIFE DST SRC CMD STS TNS 0-ADRESSENDATEN
LESEN DST SRC CMD STS TNS 1-ADRESSENDATEN
EINGABE 2-ADRESSENKANAL

2:2 ANTWORTEN

LESEN, FEHLERSUCHEN, EINGABE UND SCHLEIFE:
SRC DST CMD STS TNS DATEN

SCHREIBEN UND BIT:
SRC DST CMD STS TNS

DST IST DIE STATIONSNUMMER DER STEUERGERÄT-SCHNITTSTELLE,
DIE EIN KOMMANDO EINLEITET.

SRC IST DIE ADRESSE DER STEUERGERÄT-SCHNITTSTELLE, DIE
AUF EIN KOMMANDO ANTWORTET.

CMD IST EIN 8-BIT-WERT MIT:

BIT 7 0
BIT 6 0=KOMMANDO
1=ANTWORT
BIT 5 0=NORMAL
1=PRIORITÄT
BIT 4 0

BITS 3-0:

0- SCHREIBEN
1- LESEN
2- BITSTEUERUNG
3- PRIVILEGIERTES SCHREIBEN
4- PRIVILEGIERTES LESEN
5- PRIVILEGIERTE BITSTEUERUNG
6- FEHLERSUCHE

STS IST NULL FÜR ALLE KOMMANDONACHRICHTEN
BITS 3-0 = ORTSFEHLER:

0 - KEIN FEHLER
2 - NICHT ZUSTELLBARE NACHRICHT

BITS 7-4 = FERNFEHLER (WENN ORTSFEHLER =0)

- 0 - KEIN FEHLER
- 1 - UNZULÄSSIGES KOMMANDO ODER GRÖSSE
- 2 - PLC-SPEICHER NICHT VERIFIZIERT
- 3 - PLC NICHT VERBUNDEN
- 4 - PLC-KOMMUNIKATIONS-FEHLER
- 5 - PRIVILEGIERTER EINGRIFF
- 6 - PLC IN PROGRAMM-BETRIEBSART

TNS IST EIN 8-BIT-WERT, DER VON DER KOMMANDO-NACHRICHT IN DIE ANTWORT KOPIERT WIRD. ER KANN AUF IRGENDNEINEN WERT EINGESTELLT SEIN UND SOLL FÜR ZUGEORDNETE ANTWORTEN MIT AUSSTEHENDEN KOMMANDOS ALS MARKIERFELD DIENEN.

ADDR(ADRESSE) IST EINE 16-BIT-ADRESSE

DATA(DATEN) IST EIN FELD MIT DATEN-BYTES

PORT(KANAL) IST EINE 8-BIT-E/A-KANALNUMMER

MASKS(MASKEN) SIND 1 BIS 35 MASKEN IN DER FORM VON:

ADDR SET RESET

SET IST EINE 16-BIT-MASKE MIT BITS ZUM SETZEN

RESET IST EINE 16-BIT-MASKE MIT BITS ZUM RÜCKSETZEN.

3.0 VIelfachpunkt-Datenverbindungs-Formate

NACHRICHT:	FLAG	DST	STS	DATA	CRC	FLAG
STATUS:	FLAG	SRC	DST	STS	CRC	FLAG
AUFRUFEN:	FLAG	255	P	FS	M	CRC FLAG
ANTWORT:						

EINE AUFRUFANTWORT IST EIN TRÄGER, DER WÄHREND EINES AUF EIN AUFRUFKOMMANDO FOLGENDEN INTERVALLS AUSGESENDET WIRD. JEDER AUFRUF NIMMT ETWA 3 MILLISEKUNDEN IN ANSPRUCH. DAS INTERVALL ZWISCHEN DEN NACHRICHTEN BETRÄGT ETWA 0,5 MILLISEKUNDEN.

FLAG IST EIN HARDWARE-ERZEUGTES RAHMENZEICHEN

DST IST DIE NUMMER DER HAUPTSTATION

SRC IST DIE NUMMER DER NEBENSTATION

DATA SIND BIS ZU 252 BYTES AN DATEN

CRC IST EIN HARDWARE-ERZEUGTES PRÜFPOLYNOM

STS IST EIN 8-BIT-WERT MIT:

BIT 7 1

BIT 6 1=NEBENSTATIONSPUFFER VOLL

BITS 5-0 FEHLERCODE

- 0 - KEIN FEHLER
- 16 - SCHLECHTES RAHMENFORMAT

255 IST DIE GLOBALNETZWERKADRESSE

P IST DER PRIORITÄTSPEGEL (0 ODER 1)

FS IST DIE ERSTE ADRESSE EINES GRUPPENAUFRAFES

M IST DIE GRÖSSE EINES GRUPPENAUFRAFES

Patentansprüche

1. Verfahren zur Steuerung der Nachrichtenübertragung in einem Kommunikationsnetzwerk mit einer Vielzahl an ein Kabel angeschlossener Schnittstellen-Module, denen jeweils ein elektrisches Gerät zugeordnet ist und von denen jeder in der Lage ist, zur Steuerung der Übertragung von das zugeordnete elektrische Gerät betreffenden Datennachrichten über das Kabel die Kontrolle über das Kommunikationsnetzwerk zu erlangen und nach der Beendigung der Übertragung der Datennachrichten über das Kabel die Kontrolle

über das Kommunikationsnetzwerk an einen anderen der Schnittstellen-Module abzugeben, dadurch gekennzeichnet, daß ein von der Vielzahl der Schnittstellen-Module (8, 10, 13, 14) die Kontrolle über das Kommunikationsnetzwerk innehabender Schnittstellen-Modul zur Abgabe dieser Kontrolle

- in einem ersten Suchschritt eine an die Schnittstellen-Module gerichtete Aufrufkommandonachricht sendet, worauf sendewillige Schnittstellen-Module Antwortsignale auf das Kabel (1) legen, worauf
- in einen zweiten Suchschritt an eine Hälfte der Schnittstellen-Module wiederum eine Aufrufkommandonachricht gesendet wird, worauf bei einem Antwortsignal an eine Hälfte aus dieser Hälfte als Suchschritt eine Aufrufkommandonachricht gesendet wird und weitere Schritte bei Antwortsignalen sich anschließen, bis ein sendewilliger, die Kontrolle übernehmender Schnittstellen-Modul gefunden ist, oder andernfalls beim Ausbleiben eines Antwortsignals die Suchschritte mit einer Hälfte der jeweiligen anderen Hälfte fortgesetzt werden, bei welchen Schritten die Hälften nach einer Priorität vorgewählt werden.

2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, daß der die Aufrufkommandonachrichten aussendende Schnittstellen-Modul im Anschluß an das Aussenden der an einen einzigen sendewilligen Schnittstellen-Modul gerichteten Aufrufkommandonachricht durch das von diesem sendewilligen Schnittstellen-Modul abgegebene Antwortsignal unter Aufgabe der Kontrolle über das Kommunikationsnetzwerk zum Empfangen von Datennachrichten freigegeben wird.

3. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, daß jeder Schnittstellen-Modul eine eindeutige Stationsnummer hat und zum Erlangen der Kontrolle über das Kommunikationsnetzwerk auf eine seine Stationsnummer umfassende Aufrufkommandonachricht anspricht und daß die von dem abgabewilligen Schnittstellen-Modul aufeinanderfolgend ausgesendeten Aufrufkommandonachrichten zunehmend weniger Stationen umfassen und die letzte Aufrufkommandonachricht nur noch eine einzige Stationsnummer enthält.

4. Verfahren nach einem der vorstehenden Ansprüche, dadurch gekennzeichnet, daß ein sendewilliger Schnittstellen-Modul aufgrund der Erfassung einer an ihn allein gerichteten Aufrufkommandonachricht die Kontrolle über das Kommunikationsnetzwerk übernimmt.

5. Verfahren nach einem der vorstehenden Ansprüche, dadurch gekennzeichnet, daß die ausgesendete Aufrufkommandonachricht eine Prioritätsaufrufkommandonachricht ist, auf die nur solche sendewilligen Schnittstellen-Module antworten, die zum Aussenden einer Prioritätsdatennachricht das Kommando über das Kommunikationsnetzwerk erlangen wollen.

6. Schnittstellen-Modul für ein Kommunikationsnetzwerk, in dem die Nachrichtenübertragung gemäß dem Verfahren nach einem der Ansprüche 1 bis 5 gesteuert wird, gekennzeichnet durch eine Einrichtung (25, 29), die in Abhängigkeit vom Auftreten oder Ausbleiben eines Antwortsignals aufgrund einer ausgesendeten Aufrufkommandonachricht das Aussenden einer weiteren Aufrufkommandonachricht veranlaßt, die an eine geringere Anzahl von Schnittstellen-Modulen gerichtet ist.

7. Schnittstellen-Modul für ein Kommunikationsnetzwerk, in dem die Nachrichtenübertragung gemäß dem Verfahren nach einem der Ansprüche 1 bis 5 gesteuert wird, gekennzeichnet durch eine Einrichtung (25, 29), die in Abhängigkeit vom Auftreten oder Ausbleiben eines Antwortsignals aufgrund einer ausgesendeten Aufrufkommandonachricht das Aussenden einer weiteren Aufrufkommandonachricht veranlaßt, die bezüglich eines die Schnittstellen-Module kennzeichnenden Parameters modifiziert ist.

Hierzu 17 Blatt Zeichnungen

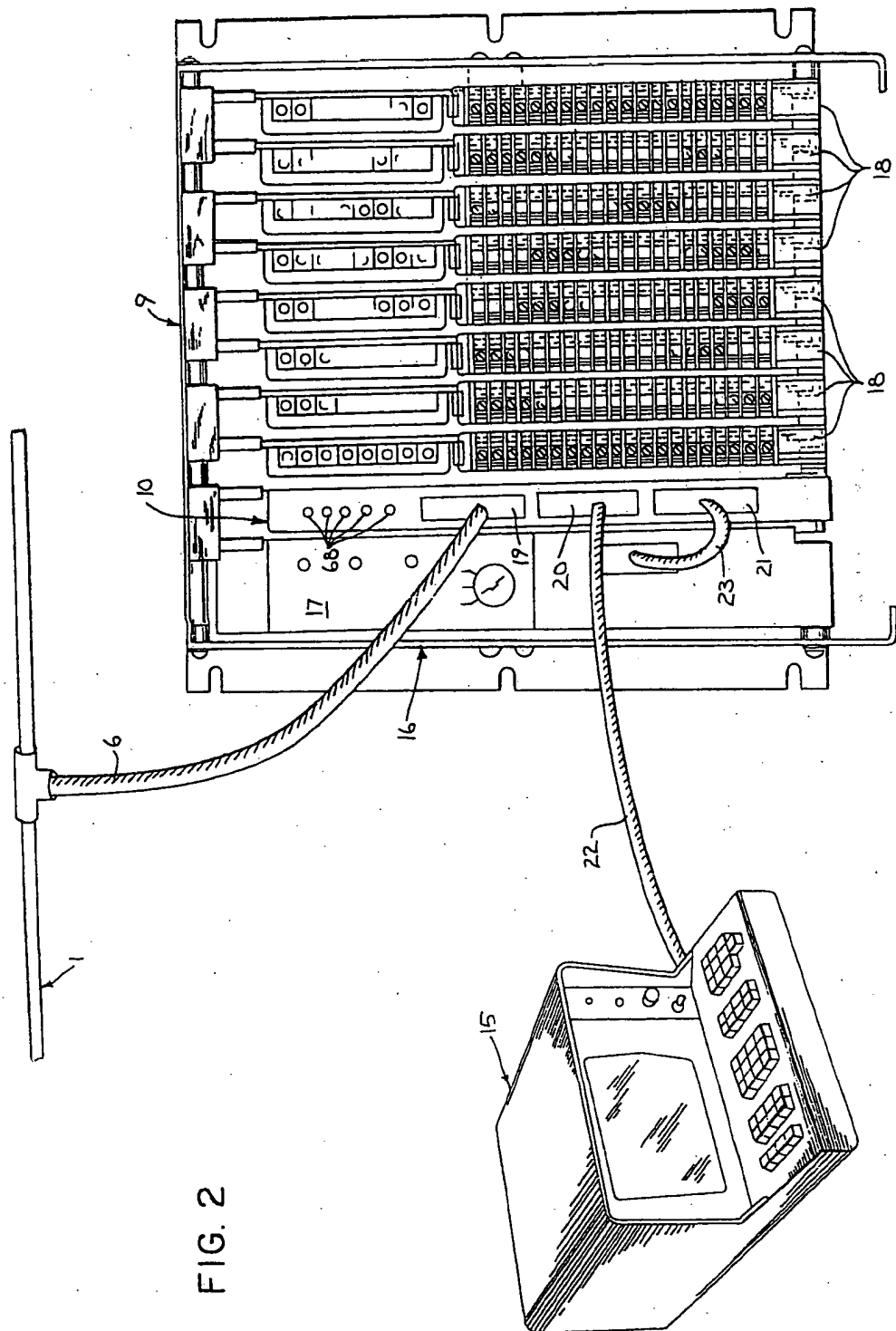


FIG. 2

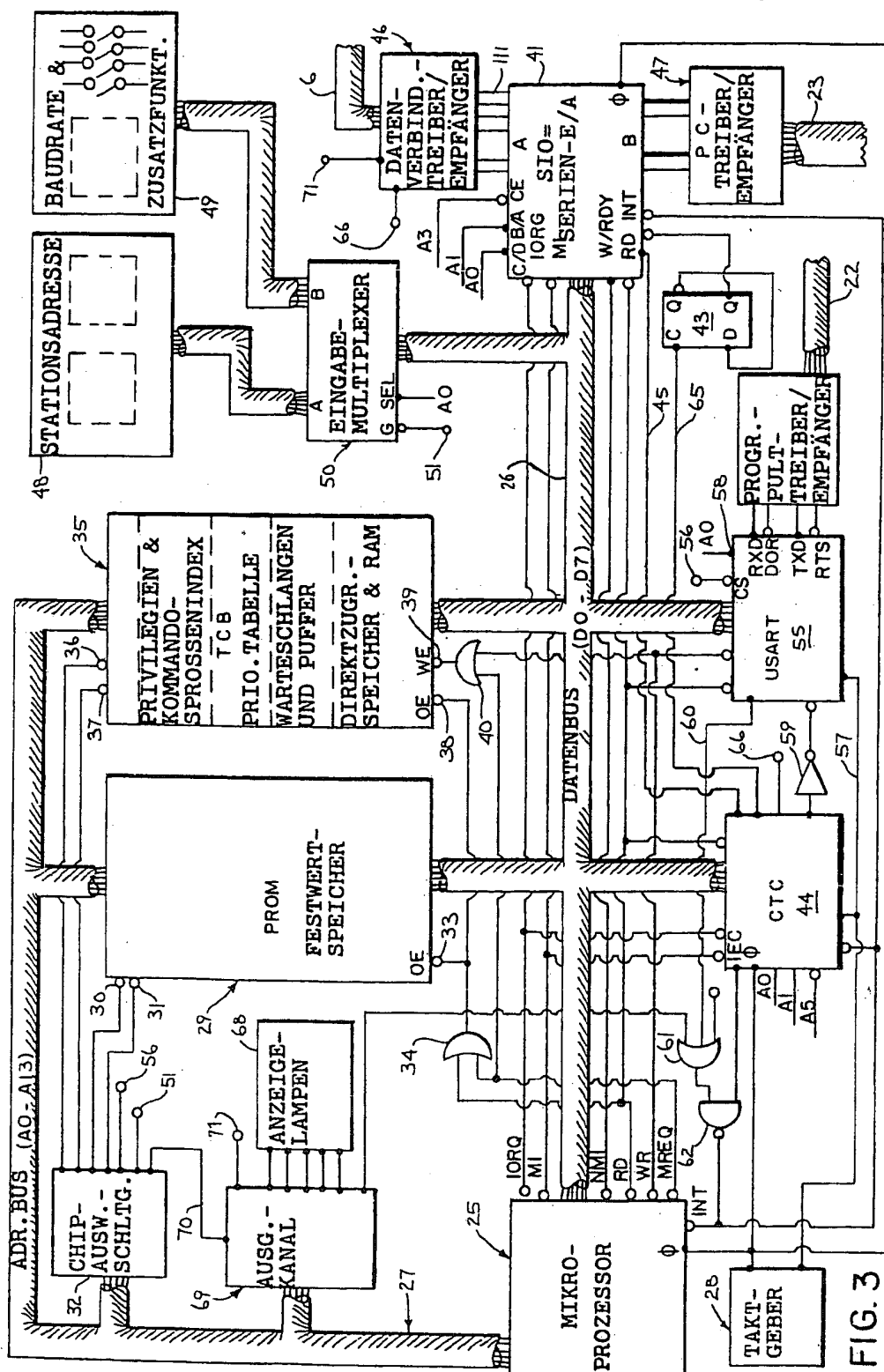
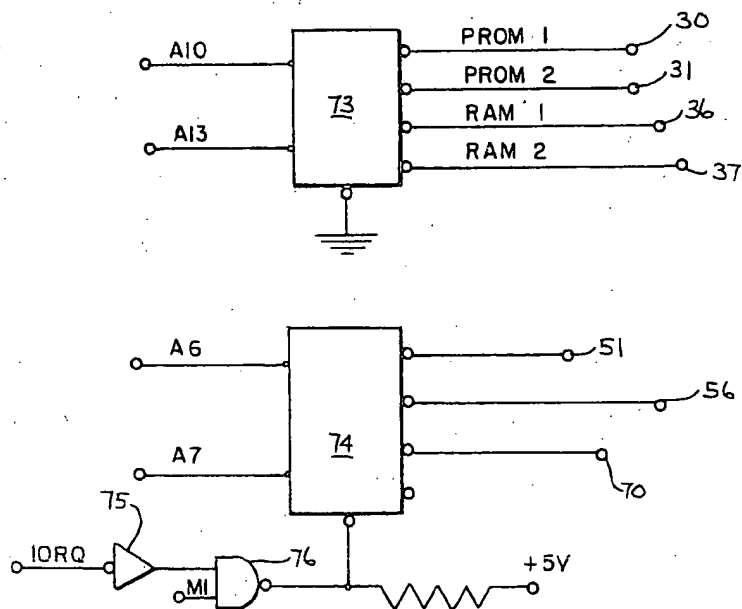


FIG. 4



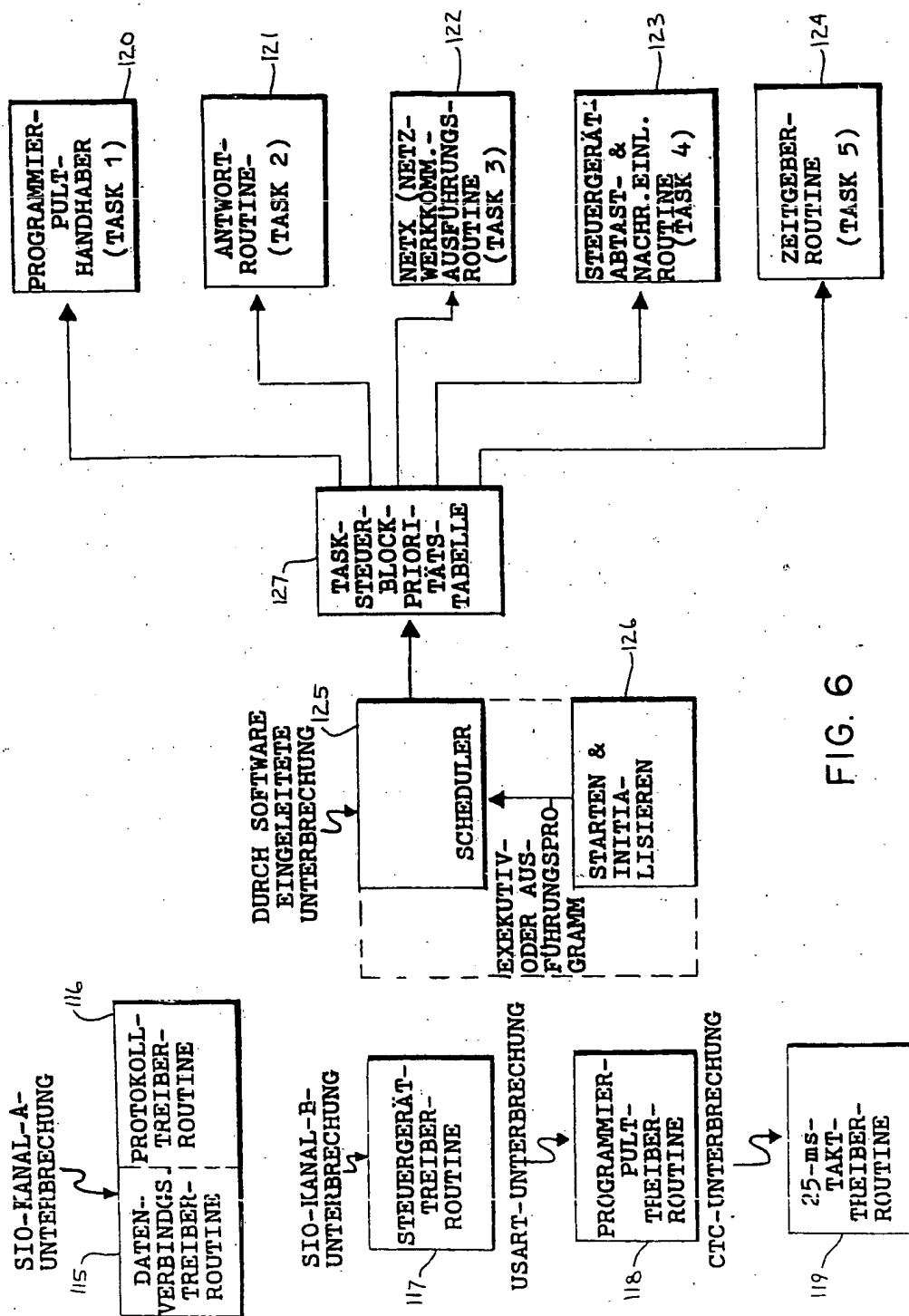


FIG. 6

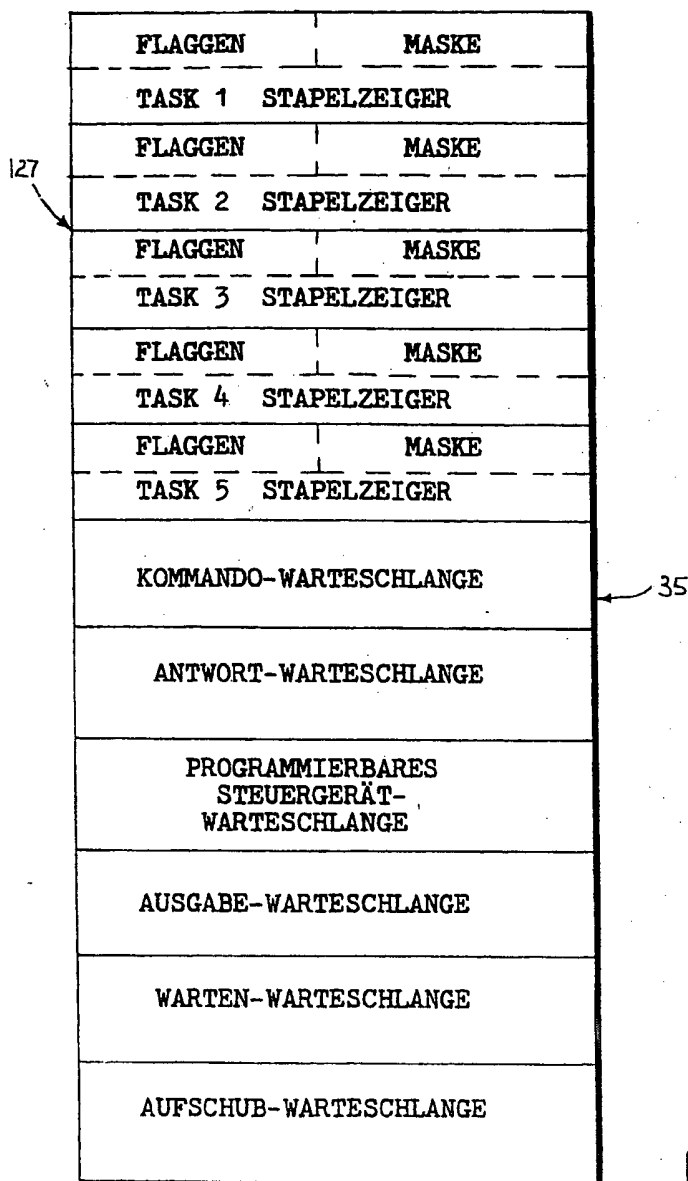


FIG. 7

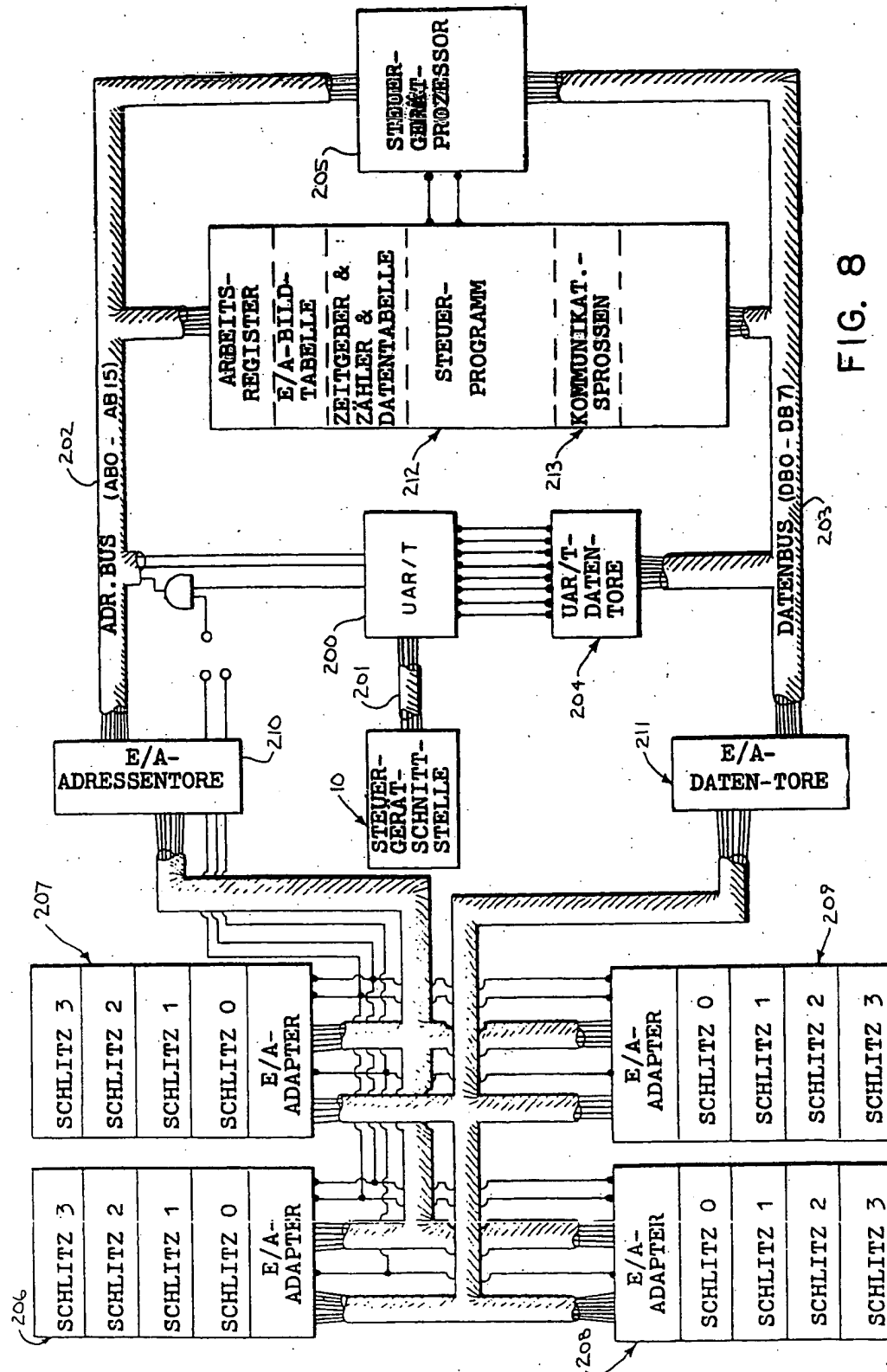


FIG. 8

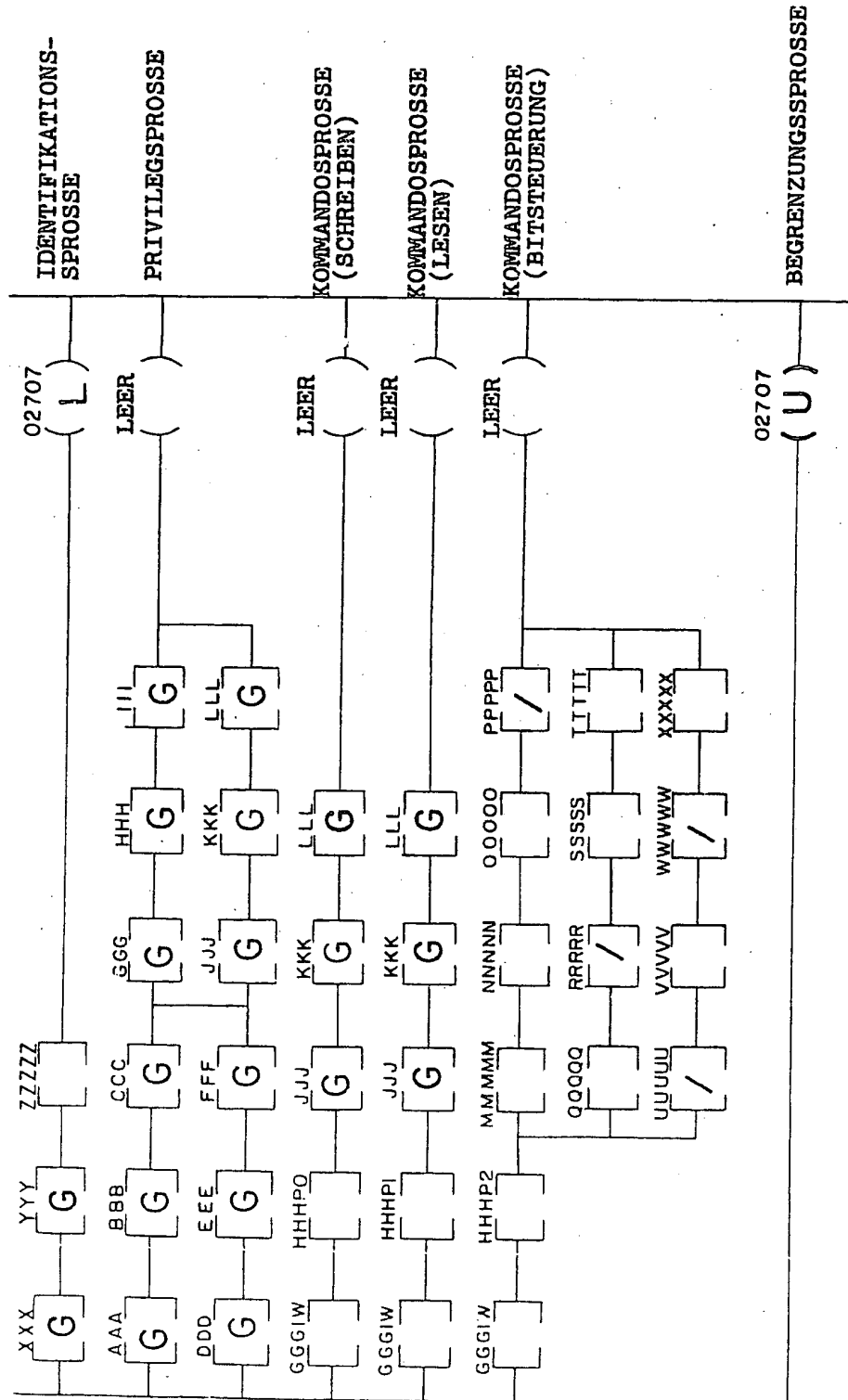


FIG. 9

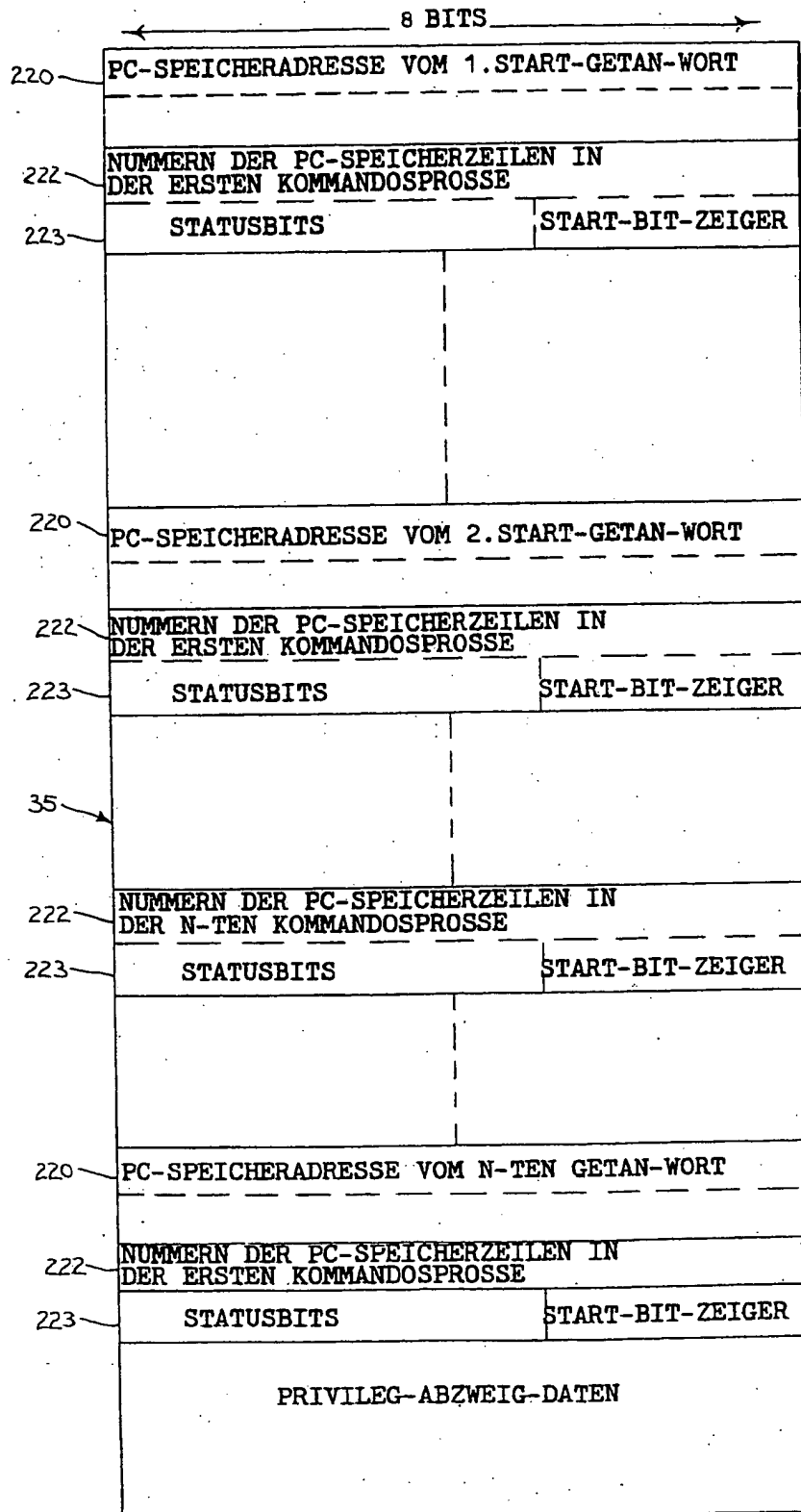


FIG.10

PRIVILEG- UND KOMMANDOSPROSSENINDEX

FIG. II

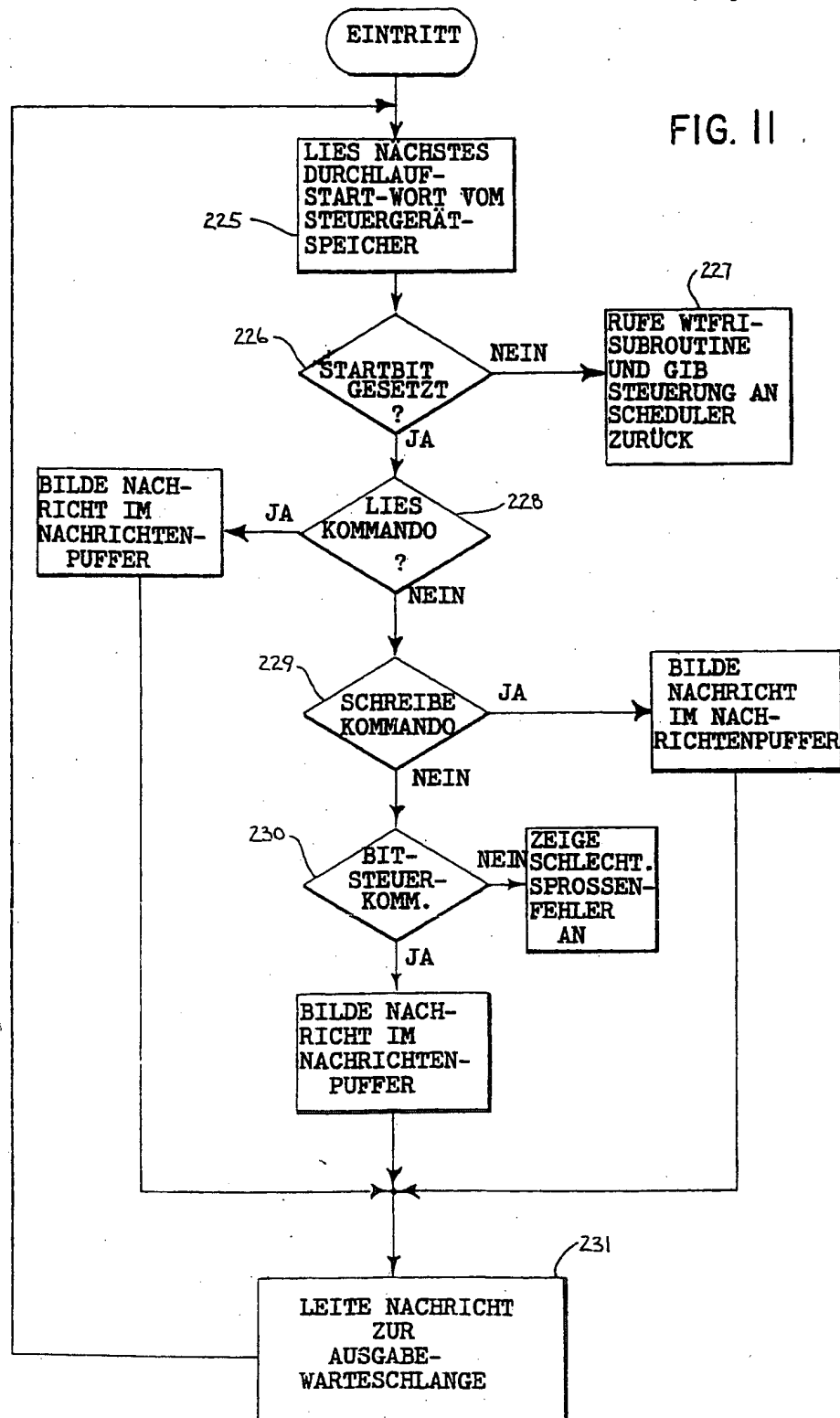


FIG. 12

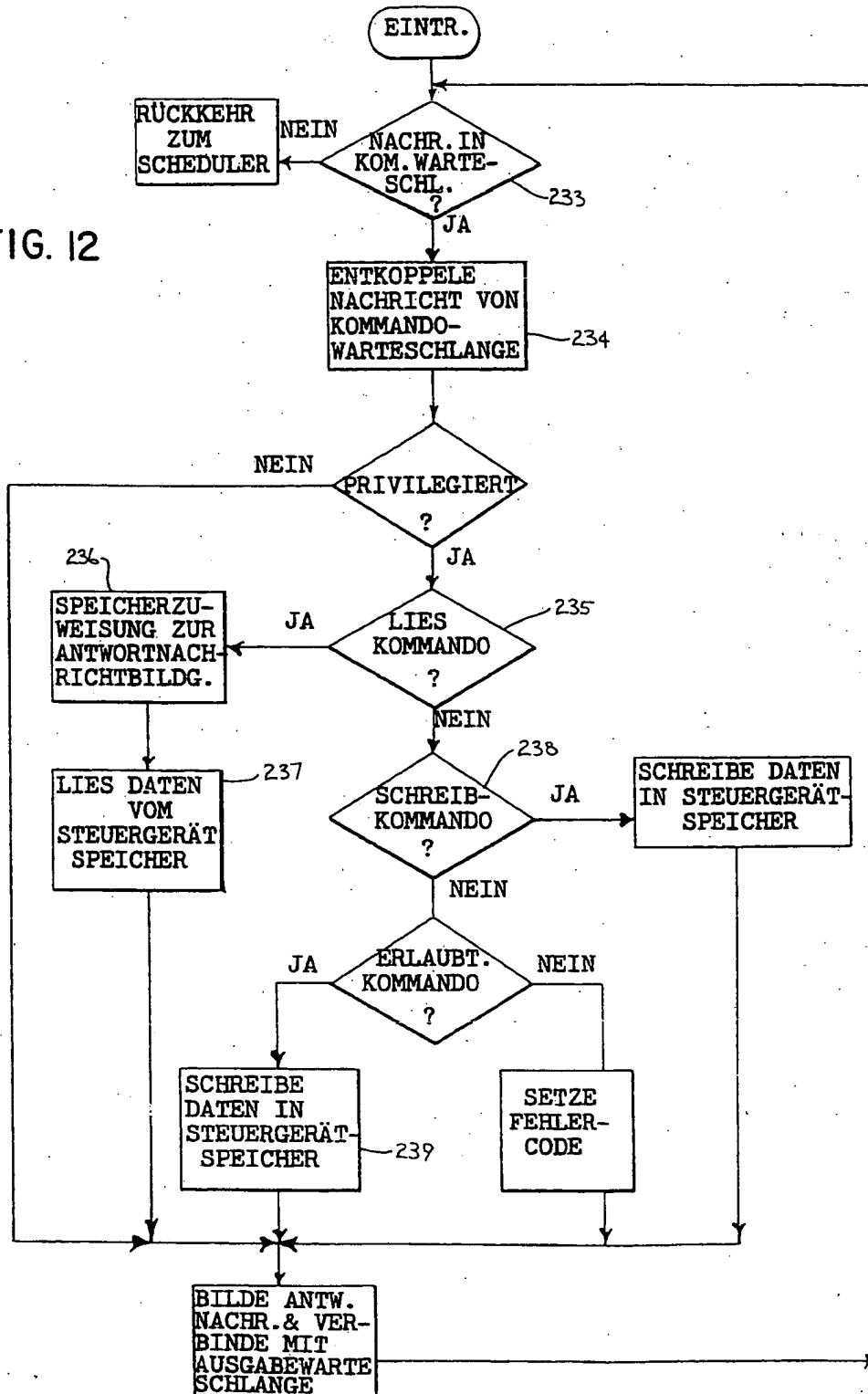


FIG. 13A

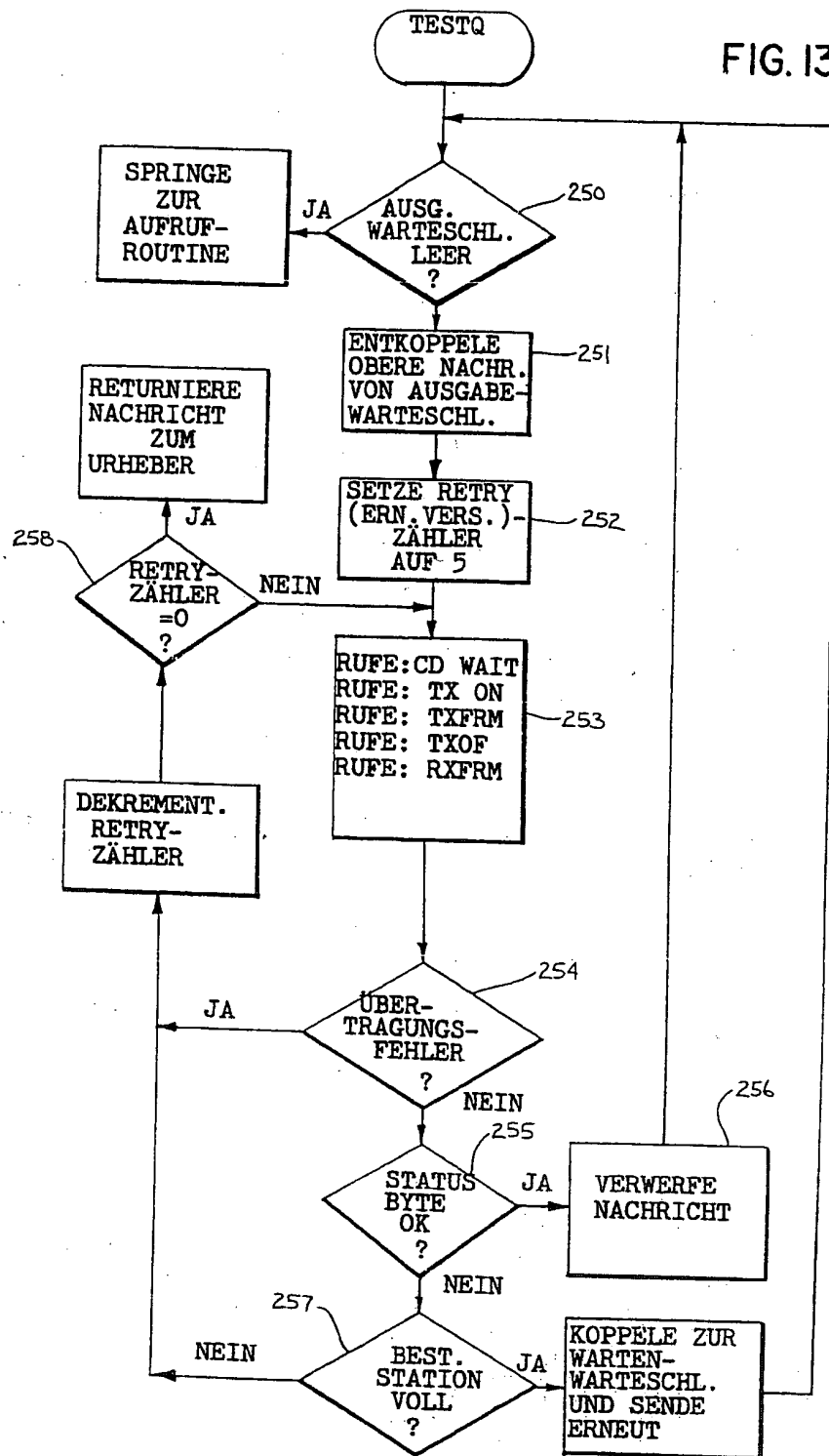


FIG. 13B

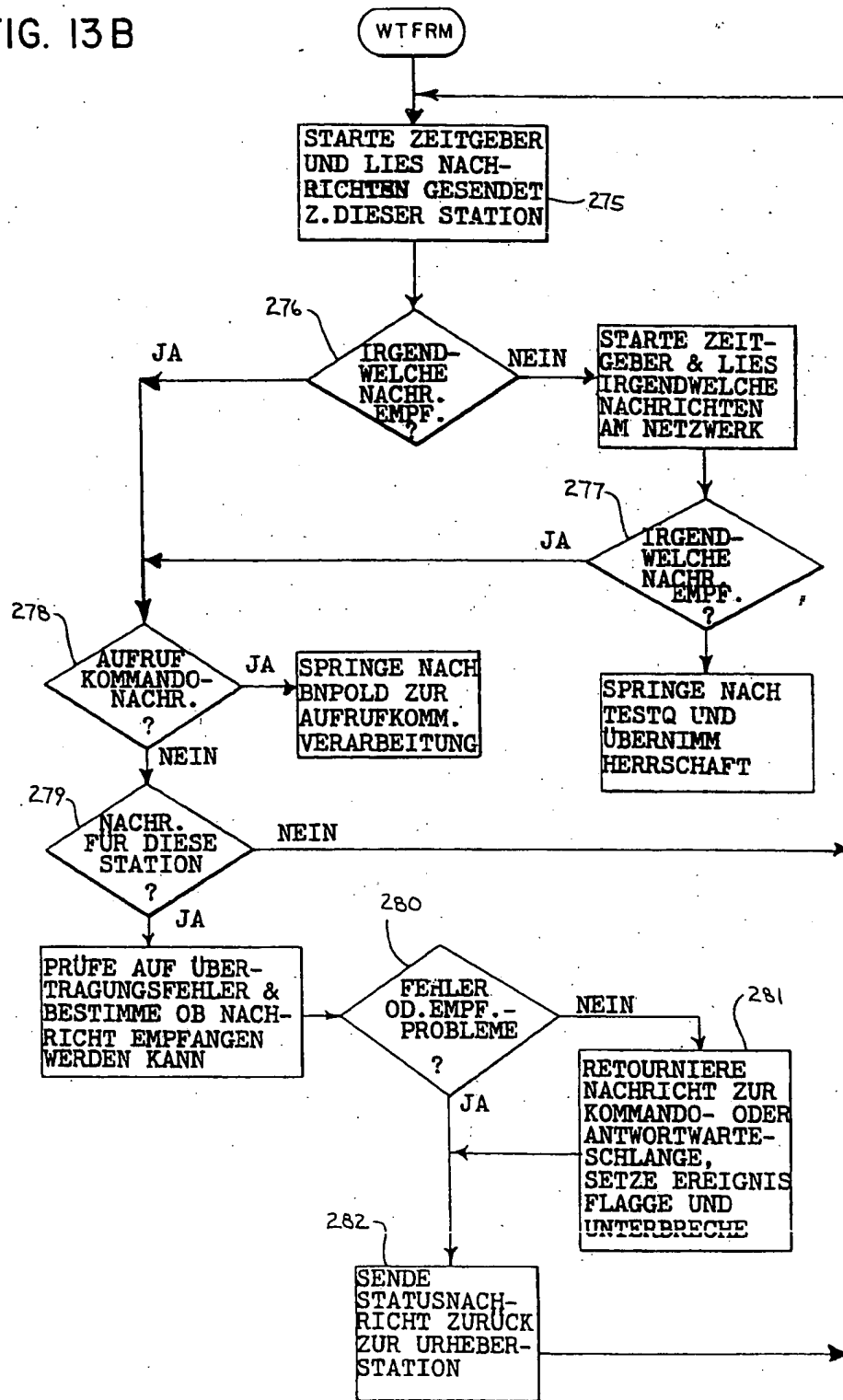
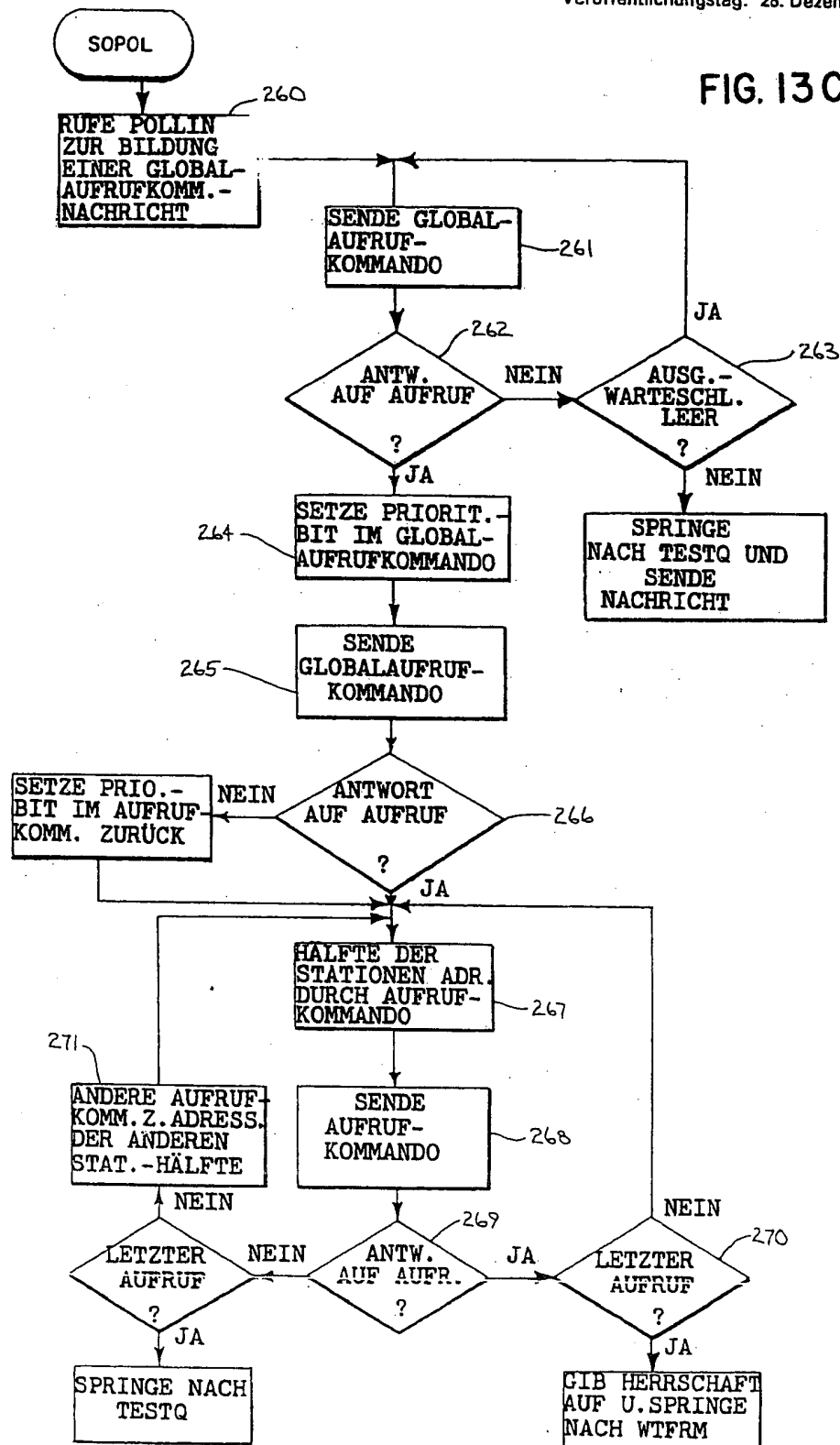


FIG. 13C



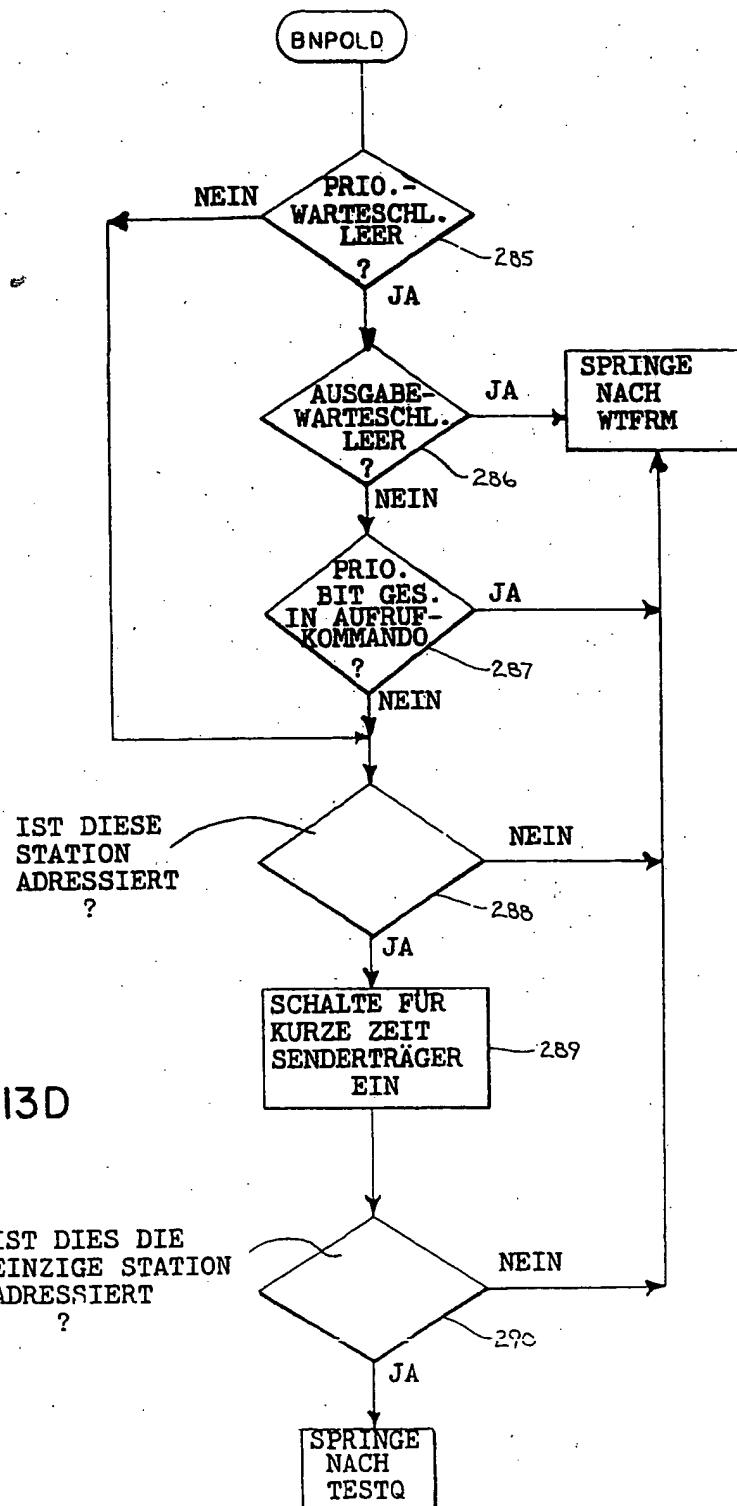


FIG. 13D

